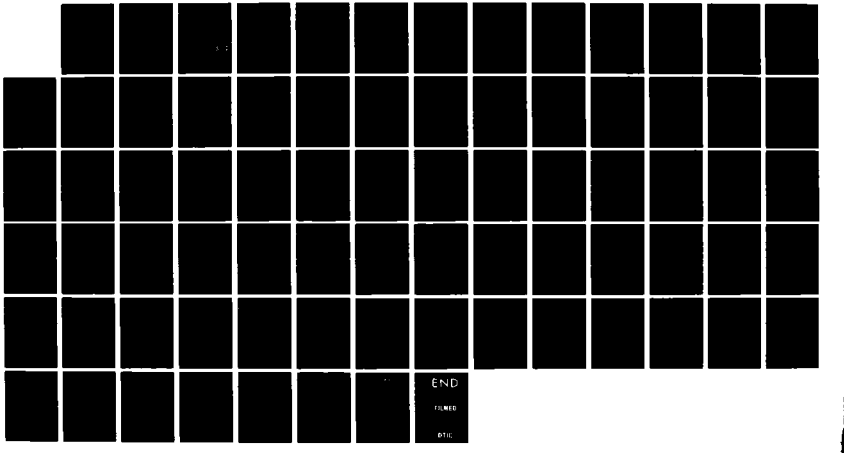
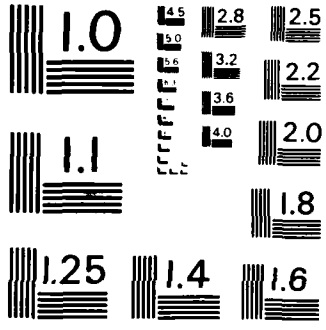
AD-A161 636 PROTOTYPE IMPLEMENTATION OF COSTCASTER A COST-PREDICTION AND TRADE-OFF MO..(U) DESMATICS INC STATE COLLEGE PA G J ZUNIC ET AL. JUL 85 TR-110-8 1/1

UNCLASSIFIED F33600-82-C-0466 F/G 5/3 NL





PROTOTYPE IMPLEMENTATION OF COSTCASTER, A COST-PREDICTION AND TRADE-OFF MODEL FOR AIR FORCE GROUND C-E EQUIPMENT

by

Gregory J. Zunic
Kevin C. Burns
Robert L. Gardner
Dennis E. Smith

— STATISTICS —

— OPERATIONS RESEARCH —

— MATHEMATICS —

DESMATICS, INC.
P.O. Box 618
State College, PA 16804

DESMATICS, INC.

P. O. Box 618
State College, PA 16804
Phone: (814) 238-9621

Applied Research in Statistics - Mathematics - Operations Research

PROTOTYPE IMPLEMENTATION OF COSTCASTER, A COST-PREDICTION AND TRADE-OFF MODEL FOR AIR FORCE GROUND C-E EQUIPMENT

by

Gregory J. Zunic
Kevin C. Burns
Robert L. Gardner
Dennis E. Smith

TECHNICAL REPORT NO. 118-8

Original Draft April 1985
Final ~~Draft~~ July 1985

Prepared under Contract No. F33600-82-C-0466

EXECUTIVE SUMMARY

COSTCASTER is a computer-based system developed by Desmatics, Inc. for the U.S. Air Force. Its purpose is to aid in making cost-related decisions regarding replacement, modification or retention of Air Force ground Communications-Electronics (C-E) equipments. This report includes a brief summary of the COSTCASTER methodology, which is described more fully in a previous Desmatics report. However, the major emphasis here is to document the implementation of a prototype computer model developed to demonstrate the feasibility of the COST-CASTER concept. Diagrams are included to show the relationships among the major components of the prototype computer model. An example is presented to illustrate the use of the model in realistic situations.

COSTCASTER is designed to use a data base of Operating and Support (O&S) costs derived from the C-E subsystem (D160A) of the Air Force Visibility and Management of Operating and Support Costs (VAMOSC) system. The COSTCASTER data base will be updated annually by a processing step which not only extends the time frame of the data base, but also computes preliminary estimates of the modification/replacement potential of C-E end items. These estimates aid in identifying candidates for further consideration.

COSTCASTER has a cost prediction submodel which applies statistical regression techniques to historical costs in order to predict costs in future years. The user may specify which cost categories to include, how many years to forecast and how to weight each year's

historical costs.

The most significant feature of COSTCASTER is the trade-off assessment submodel, which may be used to aid in assessing the consequences of possible modification/replacement/retention alternatives. COSTCASTER employs a model which considers the predicted cost of the present item over the remainder of its economic life, as compared with the anticipated acquisition cost and expected O&S cost of a proposed replacement item.

The user may achieve a high degree of interaction with the trade-off module. The model allows specification of expected economic life (both for old and new items), expected acquisition costs for replacement items, expected percentage reductions in O&S costs (by category) for new or modified items, and discount rates. Default values are provided if no user options are given. The user may try an alternative, observe its economic consequences and try additional alternatives as desired.

Most of the planned features of the COSTCASTER model have been implemented in a prototype program on a large mainframe computer for a small data base. This has established that the theoretical concepts embodied in COSTCASTER can be realized in an interactive computer environment. Desmatics is currently developing a version of COSTCASTER using Lotus 1-2-3 on a Z-100 microcomputer. This revised version will be discussed in a forthcoming technical report.

TABLE OF CONTENTS

I. INTRODUCTION

This report describes the prototype version of COSTCASTER, a cost-prediction and trade-off model being developed by Desmatics, Inc. under Air Force Contract No. F33600-82-C-0466. COSTCASTER is intended for use as a decision aid in determining whether to modify, replace or retain Air Force ground Communications-Electronics (C-E) equipment. Specifically, the COSTCASTER computer programs developed by Desmatics are designed to provide the user with forecasts of future O&S costs at the TMS level and estimates of savings associated with potential modification/replacement decisions. These forecasts will be based on historical O&S cost data obtained from the C-E subsystem of the Air Force VAMOSC system.

The methodology underlying COSTCASTER is described in detail in Desmatics Technical Report No. 118-4 [1]. (The reader may find it useful to review that report.) This report describes prototype computer programs which incorporate that methodology. The current model consists of two primary modules, one interactive and one which is designed as a batch program. There are, in addition, three peripheral graphics programs which may be used to display outputs produced by the interactive portion of the model.

Section II of this report discusses the role of COSTCASTER, the current status of the data base, and the program inputs and outputs. Section III shows how COSTCASTER might be used to perform trade-off analysis on an actual TMS and provides sample program outputs. Section

IV provides a ~~detailed~~ description of the program logic for the interactive module. The source listing for the interactive portion of the program is given in the appendix.

The programs described in this report are prototypes developed to demonstrate the feasibility of implementing the COSTCASTER methodology. They have been implemented on a large IBM mainframe computer using a small data base. Desmatics is also tasked with examining the feasibility of implementing COSTCASTER on a microcomputer. As part of this examination, Desmatics is developing a version using Lotus 1-2-3 on a Z-100 microcomputer. This revised version of COSTCASTER will be discussed in a forthcoming technical report.

II. PROGRAM DESCRIPTION

Desmatics, Inc. has written prototype computer programs to produce most of the COSTCASTER outputs described in Desmatics Technical Report No. 118-4 [1]. Currently, the programs produce ten output products: seven tables and three graphs. This section provides a brief description of the programs and their intended role in the decision-making process. The current status of the data base is outlined along with the program inputs and outputs. A detailed example is provided in Section III.

A. THE ROLE OF COSTCASTER

COSTCASTER is of value to the cost analyst for two reasons. First, COSTCASTER gives the analyst forecasts of future O&S costs for a TMS based on historical O&S costs, and provides a measure of the accuracy and reliability of these forecasts by calculating prediction intervals and prediction diagnostics.

Second, COSTCASTER provides the analyst with a means of comparing the remaining life cycle costs of a current end item with the life cycle costs of a particular modification or replacement item. By using COSTCASTER, an analyst can readily estimate the expected total savings over the remaining economic life of the current end item, as well as the expected short term cost avoidance associated with modification or replacement. These estimates can be obtained for various discount rates,

acquisition costs, and economic lifetimes. COSTCASTER also enables the cost analyst to estimate the cost associated with delaying a modification/replacement decision for various discount rates, acquisition costs, and economic lifetimes.

In short, COSTCASTER allows the analyst to see quickly what effect different assumptions have on the life cycle costs of ground C-E equipment. This knowledge of the life cycle costs of a number of alternative end items, combined with an understanding of differences in operational effectiveness, will be of great use to the analyst in the performance of trade-off analysis.

B. CURRENT STATUS OF THE DATA BASE

C-E data is currently available for only two years. At least three years of data are needed to implement COSTCASTER. Therefore, the examples given in this report are not based on data obtained from the C-E system. Instead, they are based on data obtained from the Logistics Support Cost (LSC) model.

The LSC model, which is described in AFR 400-63 Vol. 2, contains only five cost categories, instead of the nineteen categories provided by the C-E system. These five categories are: base maintenance personnel, maintenance materiel, transportation and packaging, depot maintenance, and replacement investment. However, the COSTCASTER programs are capable of analyzing data from any group of cost categories, as well as a single cost category, provided the data is available.

C. USER INPUTS

The noninteractive portion of COSTCASTER calculates break-even reduction points (BERPs), which provide preliminary indications of the modification/replacement potential of C-E end items. This portion of the model has no user inputs. In practice, an analyst will be working with a copy of the BERP tables, rather than the actual program which need be run only once each year, producing hard-copy outputs, microfiche, or a disk file for on-line access.

User inputs for the interactive portion of COSTCASTER are specified via two input screens, which display a series of questions and default responses regarding user assumptions. The first screen (Figure 1) gives the user the option of specifying the following inputs:

1) the TMS to be analyzed,

2) the discount rate (if not specified, 10% is used),

3) the acquisition cost of the modified/replacement item (if not specified, the unit price of the current TMS is used),

4) the remaining economic life of the current TMS (if not specified, 10 years is used),

5) the economic life of the modified/replacement item (if not specified, 10 years is used),

and 6) the weights to be used in the prediction submodel (if not specified, equal weights are used).

The analyst also has the option of requesting the same inputs as the last time the program was used. If the analyst requests the same inputs as last time, this screen is redisplayed with the old inputs. The

```
                    *** COSTCASTER ***

          PLEASE ENTER THE INPUTS YOU WANT COSTCASTER TO USE FOR
          THIS ANALYSIS AND PRESS RETURN.

          WOULD YOU LIKE TO USE THE SAME INPUTS
          AS LAST SESSION?    NO

WHAT TMS WOULD YOU LIKE TO ANALYZE?    _________

WHAT DISCOUNT RATE WOULD YOU LIKE TO USE?   10%

WHAT IS THE ACQUISITION COST OF THE MOD/REPLACEMENT TMS?  $__________
(IF NOT SPECIFIED, UNIT PRICE OF CURRENT TMS USED)

WHAT IS THE REMAINING ECONOMIC LIFE OF THE CURRENT TMS?     10 YEARS

WHAT IS THE ECONOMIC LIFE OF THE MOD/REPLACEMENT TMS?       10 YEARS

TYPE OF WEIGHTING IN THE PREDICTION SUBMODEL?     EQUAL
        (EQUAL OR LINEAR)
```

Figure 1: COSTCASTER Input Screen Requesting User Assumptions. The Default Values are Displayed Where Appropriate.

user then has the option of changing these inputs, or of pressing the return key to display the second input screen.

This screen (Figure 2) gives the user the option of specifying:

1) the cost categories used for predictions and subsequent trade-off analysis (if not specified, all of the cost categories are used),

and 2) the percentage reduction in O&S costs expected to result from modification or replacement, for each of the categories used (if not specified, 50% is used for each category).

As with the first input screen, the user has the option of requesting the same inputs as were used previously. If the analyst wants to use the same inputs as last time, this screen is redisplayed with the old inputs. The user then has the option of changing these inputs, or of pressing the return key to display the COSTCASTER output menu (see Figure 3), which gives the analyst a list of outputs which can be displayed by the program. The outputs listed in the menu are obtained by pressing appropriate function keys.

D. COSTCASTER OUTPUTS

This section describes the output products currently produced by the COSTCASTER prototype program. Seven tables and three graphs may be obtained from the program. Several sample output products are given in Section III. The sample outputs illustrate the detailed example given in that section.

*** COSTCASTER ***

PLEASE INDICATE BELOW THOSE COST CATEGORIES WHICH YOU WOULD LIKE TO USE FOR THIS ANALYSIS BY PUTTING A Y(YES) OR N(NO) NEXT TO THE APPROPRIATE CATEGORY. YOU MAY ALSO INDICATE THE EXPECTED COST REDUCTION ASSOCIATED WITH THE MOD/REPLACEMENT TMS.

DO YOU WANT TO USE THE SAME CATEGORIES AS LAST SESSION? NO

COST CATEGORY		% REDUCTION	COST CATEGORY		% REDUCTION
OPERATIONS PERSONNEL	Y	60	REAL PROPERTY MAINT.	Y	60
BASE MAINTENANCE PERS.	Y	60	BASE COMMUNICATIONS	Y	60
ADMINISTRATIVE PERS.	Y	60	MEDICAL(HEALTH CARE)	Y	60
SUPPLY SUPPORT PERS.	Y	60	TDY	Y	60
FUEL	Y	60	PCS	Y	60
MAINTENANCE MATERIEL	Y	60	GENERAL DEPOT SUPPORT	Y	60
UTILITIES	Y	60	ENGINEERING SUPPORT	Y	60
DEPOT MAINTENANCE	Y	60	TRANS. AND PACKAGING	Y	60
REPLACEMENT INVESTMENT	Y	60	ADVANCED TRAINING	Y	60
BASE OPERATING SUPPORT	Y	60			

Figure 2: COSTCASTER Input Screen Requesting User Selection of Cost Categories and Percentage Reduction in Costs for Each Category Selected.

*** COSTCASTER ***

WHICH OF THE FOLLOWING COSTCASTER OUTPUTS WOULD YOU LIKE TO SEE?

PF1 - DATA INPUT TABLE

PF2 - DATA INPUT TABLE LISTED BY COST CATEGORY

PF3 - TABLE OF COST PREDICTIONS AND ASSOCIATED PREDICTION INTERVALS

PF4 - TABLE OF PREDICTION DIAGNOSTICS

PF5 - COST AVOIDANCE TABLES

PF6 - WHENBUY TABLES

PF7 - MAINTENANCE TABLES

PF8 - SAVINGS TABLE

PF9 - EXIT

Figure 3: Menu Displaying List of COSTCASTER Output Options. The Data Input Table (PF1) and Maintenance Tables (PF7) Have Not Been Implemented in This Prototype Program Because of Limitations on the Current Data Base.

1. Table of Break-Even Reduction Points

COSTCASTER produces a table containing the break-even reduction point (BERP), NSN, SRD, nomenclature, estimated ten year unit O&S cost, unit price, and average inventory for every TMS in the data base. These tables can be sorted by NSN, SRD or TMS, or by increasing BERP.

The BERP provides a preliminary estimate of the fractional reduction in total O&S costs necessary for a new or modified item to "pay for itself" over the next ten years. That is, the BERP estimates the reduction needed for the ten-year savings in O&S costs to offset the acquisition cost/modification cost. The BERP may be used to identify those end items which have high O&S costs relative to their unit prices and are thus good candidates for replacement or modification. The BERP is calculated assuming that the economic lifetime of the modified or replacement end item is ten years, the remaining economic lifetime of the current end item is ten years, the discount rate is 10%, and the acquisition cost of the modified or replacement end item is the same as the unit price (last buy price) of the current item.

2. Table of Historical Costs

A second table displays the TMS's per-unit historical O&S costs, in current year dollars, for the 19 C-E cost categories over a five year period. This table enables the cost analyst to see at a glance the different costs associated with each of the 19 cost categories.

3. Predicted Costs and Associated 95% Prediction Intervals

A third table gives the user an estimate of future O&S costs for the current end item as well as 95% prediction intervals for these estimates. These intervals indicate how far actual future costs might reasonably be expected to differ from the predicted future costs. The user has the option of specifying the weights used in the prediction submodel (either equal or linear), and the remaining economic life of the current end item, which determines how many years ahead the O&S costs will be forecasted.

The predicted costs and prediction intervals can also be presented in graphical form by running a separate batch program. All of these costs are in current year dollars. Historical cost data in the COSTCASTER data base is input to the cost-prediction submodel, which employs statistical regression techniques (weighted least squares) to forecast O&S costs for future years. (The mathematical details of weighted least squares and the associated prediction intervals may be found in [2].) As specified by the user, these predictions may be made for:

(1) Total O&S costs,

(2) Costs for a single C-E O&S cost category (e.g., base maintenance personnel),

and (3) For any group of C-E O&S cost categories of interest to the user (e.g., all personnel costs).

4. Table of Prediction Diagnostics

A fourth table provides the prediction diagnostics for the per-

unit cost predictions associated with the end items. The residuals, relative accuracy index, stability index, and precision index are given.

The residuals are the differences between the observed costs and the predicted costs. Since these residuals measure the absolute accuracy of the forecasts, they tend to increase as costs increase. A more stable measure of accuracy is given by the relative accuracy index, which is the ratio of the residual to the predicted value, expressed as a percentage. Small relative accuracy indices indicate accurate forecasts.

The stability index measures the similarity of models fit in consecutive years. In particular, it measures the relative differences between their forecasts. If the cost-prediction process is performing well, then these differences will be small in absolute value.

The precision index measures the width of the prediction interval relative to the predicted value. Precision indices which are small in absolute value are indicative of more reliable forecasts.

It is very difficult to forecast costs from only a few years of historical data. Therefore, the prediction diagnostics will usually be fairly large initially. However, if the model being fit is an adequate approximation of the true cost behavior for a given TMS, these indices will tend toward zero as more historical data is obtained for that end item.

5. Table of Expected Total Savings in O&S Costs

A fifth table shows the expected total savings in O&S costs per unit during the remaining economic life of the current item for 10, 30, 50, 70, and 90% reductions in O&S costs. This table gives the expected savings as a function of the percent reduction in O&S costs and the economic life of the replacement end item. The table not only provides estimates of savings, but also makes it simple to examine the sensitivity of these estimates to the assumptions about the reduction in O&S costs and the economic life. This information can also be presented in graphical form by running a separate batch program. The user has the option of specifying the acquisition cost of the modified/replacement item, the remaining economic life of the current item, and the discount rate used in constructing the table, all of which affect the calculation of the expected savings.

6. Table of Expected Cost Avoidance

A sixth table gives the expected per unit cost avoidance for 10, 30, 50, 70, and 90% reductions in O&S costs during the remaining economic life of the current item. This table shows estimated cost avoidance as a function of time and the expected reduction in O&S costs, and provides visibility of the short-term benefits expected as a result of replacement or modification. This information can also be presented in graphical form by running a separate batch program. The user has

the option of specifying the acquisition cost of the modified/replacement item, the remaining economic life of the current end item and the discount rate used in constructing the table, all of which affect the calculation of the expected savings.

The expected cost avoidance is summed over different time periods, ranging from one year to the remaining economic life of the current end item. It is calculated as the estimated reduction in O&S costs minus the cost of modification or replacement. The total savings is the cost avoidance over the remaining life of the current item plus the residual value of the modified or replacement item.

7. Table of Expected Savings If Replacement is Delayed (WHENBUY Table)

A seventh table shows the expected per-unit savings over the life of the current TMS that would result if modification or replacement of the current end item were delayed until some future year. The user has the option of specifying the acquisition cost of the modified/replacement item, the remaining economic life of the current end item and the discount rate used in constructing the table, all of which affect the calculation of the expected savings. In addition, the user can specify the expected reduction in O&S costs for each category that will result from modifying or replacing the current end item.

III. EXAMPLE

This section provides an example illustrating the use of COSTCASTER for making cost predictions and performing trade-off analyses. O&S cost data for the AN/PRC-77 radio set is used in the illustration. This data was obtained from the LSC model [3] for FY77-FY81. The sample COSTCASTER output products for this example are given at the end of this section.

As a first step in determining whether the AN/PRC-77 radio set is a good candidate for modification or replacement, the analyst would look up this TMS in the BERP table. Table 1 provides BERPs for a sample of four C-E end items, using data from the LSC model for FY77-FY81. Of course, in the future the complete C-E data base will be used and all end items in that data base will appear in the BERP table. This table will be updated each year as new cost data becomes available.

The BERP for the AN/PRC-77 is .37, suggesting that if the O&S costs for a replacement item were at least 37% less than those for the current TMS, then replacement might be warranted from an economic standpoint. The other three end items in Table 1 have BERPs greater than one. For those items, the acquisition cost (estimated by the last buy price) is larger than the expected total ten-year O&S cost. No reduction in O&S costs could offset the expense of replacement unless the new item also had a lower acquisition cost. Of course, modification of the end item might be desirable.

The calculations used to produce the BERP table are based on a rough set of initial assumptions about the discount rate, the cost of

replacement, and the economic lifetimes of both the current and the replacement end items. These assumptions will not always be accurate. However, the BERP table is intended only as an initial screening device for identifying replacement/modification candidates. More refined assumptions are needed to obtain good estimates of the economic benefits associated with modification or replacement. COSTCASTER is designed to allow the analyst to change these assumptions and to examine the effects of these changes on the estimates produced by the model.

Historical O&S cost data for the AN/PRC-77 radio set is given in Table 2. Suppose that the cost analyst knows that this end item can be modified at a cost of $1,000 or replaced at a cost of $3,500, and wants to determine which of these options has the lowest life cycle costs relative to the AN/PRC-77. In order to make this determination, the analyst must specify the following quantities:

1) the remaining economic life of the current TMS,

2) the economic life of the replacement or modified TMS,

3) the percentage reduction in O&S costs,

and 4) the discount rate.

These quantities will seldom be known exactly but may usually be estimated with some degree of precision. By allowing the analyst to change these assumptions, COSTCASTER provides the means of determining their effects on the estimates of savings.

Suppose that the analyst estimates that the current TMS will last five more years, and that the economic lives of the modified and replacement items are ten and fifteen years respectively. The analyst also

thinks that the reduction in O&S costs for the modified end item will be at least 30% and at most 50% whereas the reduction in O&S costs for the new item will be at least 50% and at most 70%. The analyst is not sure what the discount rate will be, but thinks that it will be close to 10%.

The first thing the analyst looks at are the forecasts and associated 95% prediction intervals of future O&S costs for the radio set (Table 3 and Figure 4) and the prediction diagnostics associated with these forecasts (Table 4). (Note that the optional linear weighting scheme is used in this example.) Inspection of the forecasts reveals they are nearly constant over time, and that the prediction intervals for the forecasts are fairly narrow (the endpoints of the prediction intervals are always within $153 of the forecasts), indicating that the forecasts are probably accurate. The prediction diagnostics tend to confirm this, since the residuals, relative accuracy indices, stability indices, and precision indices are in most cases small in absolute value. (The FY80 precision index is large, but this is to be expected. That quantity measures the precision of forecasts made using only the first three years of historical data.)

Table 3 also provides a revised BERP value, based on the assumptions made by the user. It depends on the weighting scheme, the discount rate, the economic life of the replacement item, and the remaining life of the current item. In this example, only the weighting scheme has been changed from the default value and the BERP differs only slightly from that given in Table 1 (36% versus 37%).

In order to determine how the expected total cost for the modified or replacement TMS compares with the projected O&S cost for the AN/PRC-77, the analyst looks at the total savings tables for a discount rate of 10%. Table 5 shows the expected savings associated with the replacement item as a function of the percent reduction in O&S costs and the economic life of the replacement item. If this economic life is 15 years, then the expected total savings range from $573/unit (50% reduction) to $1623/unit (70% reduction), as shown in the last line of Table 5.

Table 6 shows the expected total savings associated with the modified end item. The last line in the table shows these savings for a 10 year economic lifetime. If the percent reduction in O&S costs is between 30% and 50%, then the estimated total savings are between $885/unit and $1935/unit. This range of expected savings is slightly higher than that for the replacement item, but the two ranges overlap to such an extent that it is difficult to choose between the two alternatives.

Figure 5 is a contour plot showing the savings expected to result from replacing the current item. This plot corresponds to Table 5 and allows the analyst to easily examine the sensitivity of the estimates to the assumptions about the percent reduction in O&S costs and the economic life of the replacement item.

Tables 7 and 8 show the expected short-term cost avoidance associated with the replacement and modified items, respectively. From Table 7 it can be seen that if the percentage reduction in O&S costs

for the replacement item is 50%, that item cannot be amortized during the next five years (the remaining lifetime of the current item). The corresponding contour plot (Figure 6) shows that a 66% reduction in O&S costs is necessary for the replacement acquisition cost ($3,500) to be amortized in that amount of time.

In contrast to the replacement scenario, Table 8 indicates that the modification costs ($1,000/unit) can be amortized within three years if the reduction in O&S costs is at least 30%. Modification would result in greater short-term savings than would replacement. However, the replacement item might still be a better investment because of its longer economic life (or, equivalently, greater residual value after five years).

At this point it would be clear to the analyst that either modification or replacement would result in economic benefits. However, it would not be clear which option was preferable. This uncertainty is primarily a result of not knowing precisely the percent reduction in O&S costs which would result from each of the two alternatives. Therefore, instead of considering a range of possible reductions, the analyst might specify the best available estimate for each of the two options.

Suppose the analyst thinks that replacing the AN/PRC-77 would result in a 65% reduction in base maintenance personnel costs and a 50% reduction for the other cost categories. Given these specifications, COSTCASTER automatically combines the individual estimates into an estimate of the percent reduction in total O&S costs. (For this

example, base maintenance personnel costs account for 97% of the total O&S costs, so only the reduction for that category is important.) This weighted estimate, as well as estimates of the effects of delaying the modification/replacement decision, can be obtained from the WHENBUY table.

Table 9 is a WHENBUY table for the replacement radio set, using the analyst's best estimates of economic lifetimes, percent reduction in O&S costs, discount rate, and acquisition costs. It can be seen from the table that immediate replacement would result in a savings of $1342/unit over the next five years. Delaying the decision would result in lesser, but still positive, savings.

Table 10 shows the results of delaying the modification decision. Immediate modification would result in a savings of $1410/unit over the next five years, which is slightly more than would result from replacement. In fact, at any time in the next four years, modification would be preferable to replacement. Of course, the estimated differences in savings for the two options are only slight and depend on a number of assumptions, some of which may be incorrect.

In a situation like this, where the economic benefits resulting from two options are estimated to be nearly equal, the final decision would probably hinge on other factors. Primarily, the analyst would need to know (or estimate) the relative operational effectiveness of the modified and replacement items. COSTCASTER is not designed to consider the operational effectiveness of C-E end items, but rather to deal solely with the economic factors involved in modification/replacement

decisions. Nevertheless, the forecasting and trade-off assessment capabilities of COSTCASTER make it a valuable tool for use as part of the overall decision-making process.

TABLE OF BREAK-EVEN REDUCTION POINTS FOR CE END ITEMS
IN ALPHABETICAL ORDER BY TMS.

TMS	NSN	SRD	NOMENCLATURE	BREAK-EVEN REDUCTION POINT	ESTIMATED TEN YEAR UNIT O&S COST	UNIT PRICE	AVERAGE INVENTORY
AN/MRC-85	5820006888822	7AK	RADIO_SET	6.28	63734	400000	1.00
AN/PRC-77	5820009303724	BN6	RADIO_SET	0.37	8242	3033	821.25
AN/TPS-43E	5840005489070	8K1	RADAR_SET	7.41	191146	1415678	55.75
458X24	5805004406033	CUV	MULTIPLEXER	1.02	16788	16172	6.00

Table 1: Table of Break-Even Reduction Points (BERPs)
for a Sample of Four C-E End Items.

AN/PRC-77 RADIO_SET INPUT TABLE(COSTS ARE $/UNIT IN FY82 $)

COST CATEGORY	FY77	FY78	FY79	FY80	FY81
OPERATIONS PERSONNEL	*	*	*	*	*
BASE MAINTENANCE PERS.	1381	1383	1374	1373	1384
ADMINISTRATIVE PERSONNEL	*	*	*	*	*
SUPPLY SUPPORT PERSONNEL	*	*	*	*	*
FUEL	*	*	*	*	*
MAINTENANCE MATERIEL	13	14	10	9	14
UTILITIES	*	*	*	*	*
DEPOT MAINTENANCE	69	6	61	2	6
REPLACEMENT INVESTMENT	0	0	0	0	0
BASE OPERATIONS SUPPORT	*	*	*	*	*
REAL PROPERTY MAINT.	*	*	*	*	*
COMMUNICATIONS	*	*	*	*	*
TEMPORARY DUTY	*	*	*	*	*
PERM. CHANGE OF STATION	*	*	*	*	*
MEDICAL	*	*	*	*	*
GENERAL DEPOT SUPPORT	*	*	*	*	*
ENGINEERING SUPPORT	*	*	*	*	*
TRANSPORT. AND PACKAGING	2	0	3	0	0
ADVANCED TRAINING	*	*	*	*	*

Table 2: Historical O&S Costs by Category for the AN/PRC-77 Radio Set. Asterisks are Used for Those Categories for Which the LSC Model Does Not Provide Data.

PREDICTIONS FOR THE AN/PRC-77 RADIO_SET
THE PREDICTED COST FOR YEAR T =1453*T**(-0.0238) (FY82 DOLLARS)

FY	T	COST	FORECAST	95% PREDICTION INTERVAL
77	1	1465	*	*
78	2	1403	*	*
79	3	1448	*	*
80	4	1385	*	*
81	5	1405	*	*
82	6	*	1393	(1305,1486)
83	7	*	1388	(1293,1489)
84	8	*	1383	(1282,1493)
85	9	*	1379	(1271,1496)
86	10	*	1376	(1262,1500)
87	11	*	1373	(1253,1503)
88	12	*	1370	(1246,1507)
89	13	*	1367	(1238,1510)
90	14	*	1365	(1231,1513)
91	15	*	1363	(1225,1516)

BASED ON USER INPUTS,BERP= 0.36 LINEAR WEIGHTS ARE USED.

Table 3: Cost Predictions and Associated 95% Prediction Intervals.

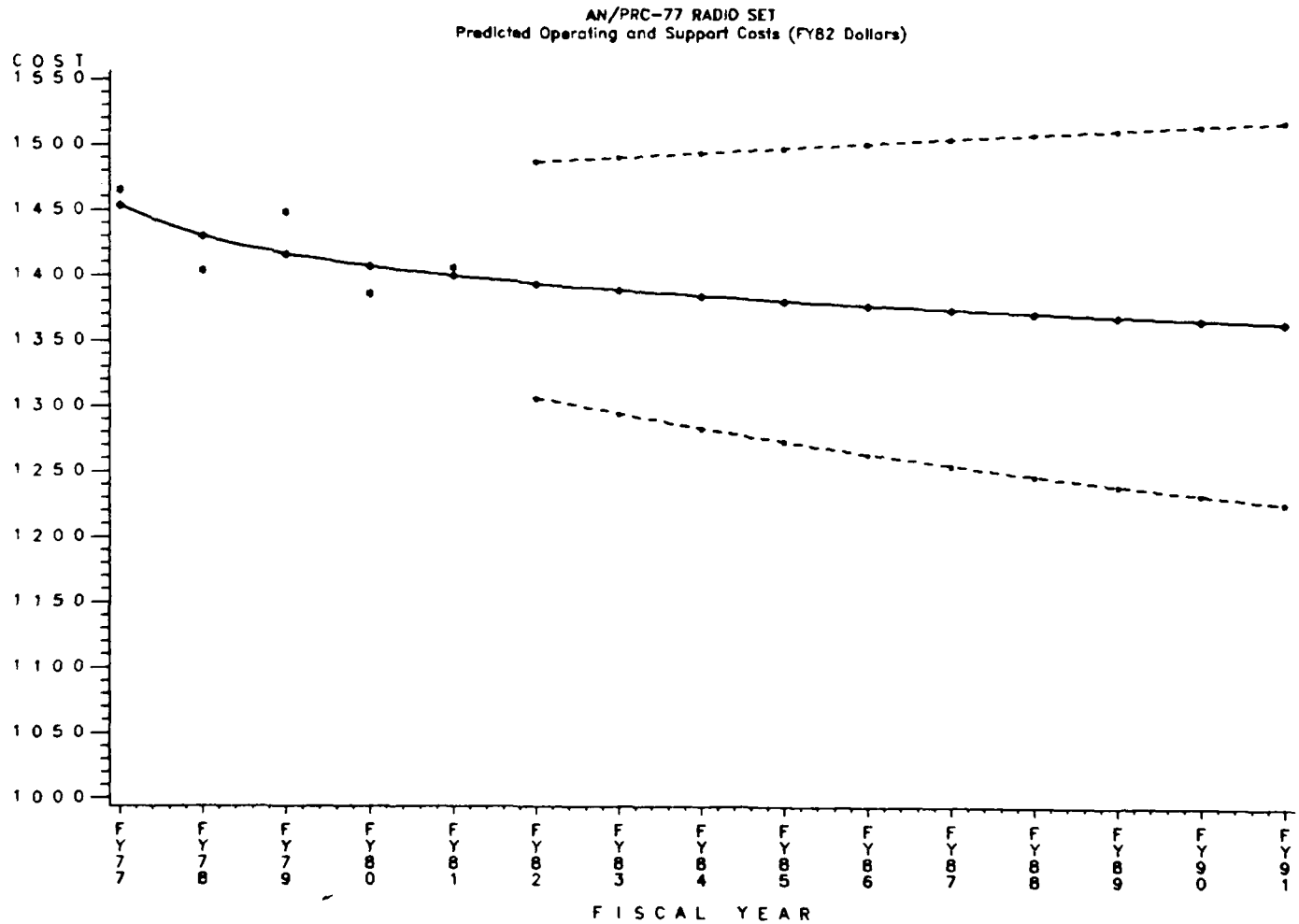


Figure 4: Graphical Presentation of Predicted Costs for the AN/PRC-77 Radio Set. Solid Curve Connects Predicted Costs. Dashed Curves Provide 95% Prediction Limits.

PREDICTION DIAGNOSTICS FOR THE AN/PRC-77 RADIO_SET

ACCURACY INDICES

FY	RESIDUAL	RELATIVE ACCURACY
80	-48	-3.3%
81	15	1.1%

FY	STABILITY INDEX
81	-2.9%
82	0.8%

FY	PRECISION INDEX
80	48.6%
81	11.8%
82	6.2%

Table 4: Prediction Diagnostics

TOTAL SAVINGS OVER THE LIFE OF ITEM REPLACING
THE AN/PRC-77 RADIO_SET
DISCOUNT RATE=10% ACQUISITION COST=$3600

LIFE OF REPLACEMENT	PERCENT REDUCTION IN SUPPORT COSTS				
	10%	30%	50%	70%	90%
5	-2976	-1926	-876	174	1223
6	-2613	-1563	-514	536	1586
7	-2364	-1306	-266	796	1844
8	-2160	-1111	-61	989	2038
9	-2009	-960	90	1140	2189
10	-1889	-839	211	1260	2310
11	-1790	-740	309	1359	2409
12	-1707	-658	392	1441	2491
13	-1638	-588	461	1511	2561
14	-1578	-528	521	1571	2620
15	-1526	-477	573	1623	2672

Table 5: Expected Savings Per Unit for the Replacement Item.

TOTAL SAVINGS OVER THE LIFE OF ITEM REPLACING
THE AN/PRC-77 RADIO_SET
DISCOUNT RATE=10% ACQUISITION COST=$1000

LIFE OF REPLACEMENT	PERCENT REDUCTION IN SUPPORT COSTS				
	10%	30%	50%	70%	90%
5	-476	574	1624	2674	3723
6	-372	678	1728	2777	3827
7	-298	752	1801	2851	3901
8	-242	807	1857	2907	3956
9	-199	850	1900	2950	3999
10	-165	885	1935	2984	4034

Table 6: Expected Savings Per Unit for the Modified Item.

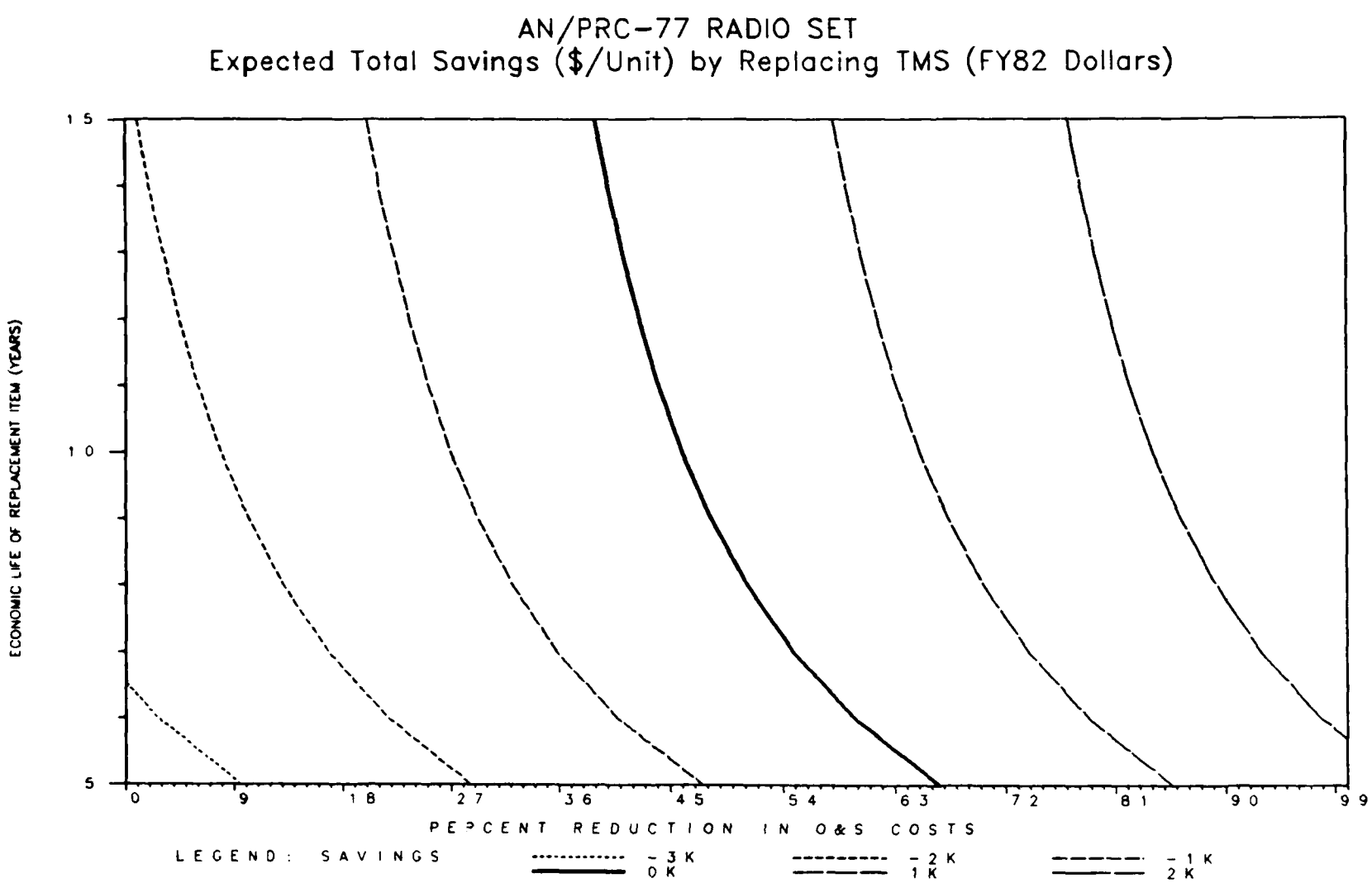


Figure 5: Expected Savings Per Unit for the Replacement Item.

EXPECTED COST AVOIDANCE AFTER N YEARS
FOR THE AN/PRC-77 RADIO_SET
DISCOUNT RATE=10% ACQUISITION COST=$3600

YEARS AFTER PURCHASE	PERCENT REDUCTION IN SUPPORT COSTS				
	10%	30%	50%	70%	90%
1	-3373	-3120	-2867	-2614	-2361
2	-3259	-2776	-2294	-1811	-1329
3	-3155	-2464	-1774	-1084	-393
4	-3061	-2182	-1303	-424	455
5	-2975	-1926	-876	174	1223

Table 7: Expected Cost Avoidance for the Replacement Item.

EXPECTED COST AVOIDANCE AFTER N YEARS
FOR THE AN/PRC-77 RADIO_SET
DISCOUNT RATE=10% ACQUISITION COST=$1000

YEARS AFTER PURCHASE	PERCENT REDUCTION IN SUPPORT COSTS				
	10%	30%	60%	70%	90%
1	-873	-620	-367	-114	139
2	-759	-276	206	689	1171
3	-666	36	726	1416	2107
4	-661	318	1197	2076	2966
6	-476	674	1624	2674	3723

Table 8: Expected Cost Avoidance for the Modified Item.

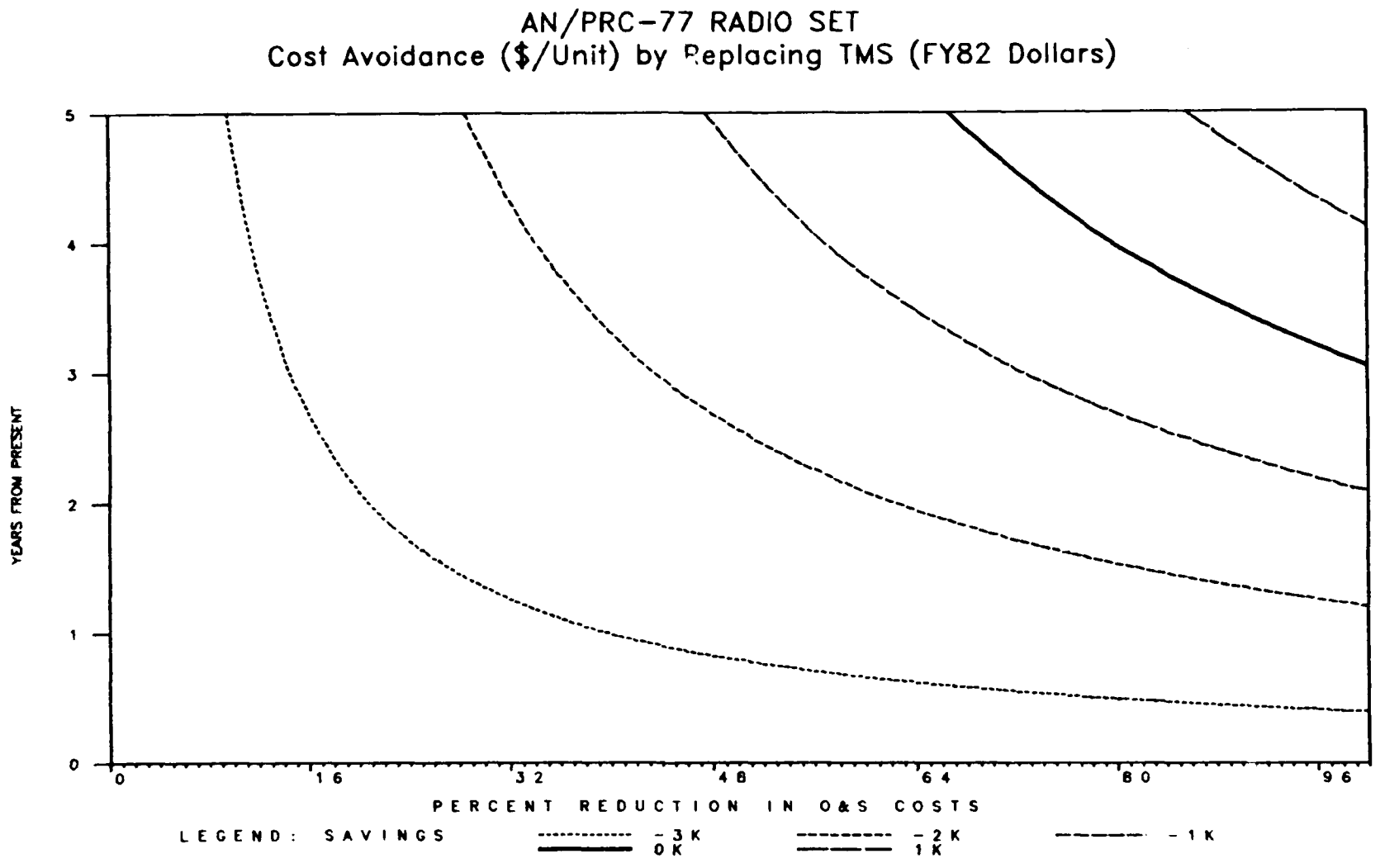


Figure 6: Expected Cost Avoidance for the Replacement Item.

EXPECTED SAVINGS IF REPLACEMENT IS MADE N YEARS FROM PRESENT
AN/PRC-77 RADIO_SET

NUMBER OF YEARS (N) TO REPLACEMENT	SAVINGS
0	1342
1	986
2	679
3	416
4	190

NOTE: SAVINGS ARE PER UNIT OVER REMAINING ECONOMIC
LIFE OF CURRENT TMS (5 YEARS)
ECONOMIC LIFE OF CURRENT TMS = 5
ECONOMIC LIFE OF REPLACEMENT TMS = 15
PERCENTAGE REDUCTION IN O&S COSTS = 66%
DISCOUNT RATE = 10%
ACQUISITION COST = 3500

Table 9: WHENBUY Table for the Replacement Item.

EXPECTED SAVINGS IF REPLACEMENT IS MADE N YEARS FROM PRESENT
AN/PRC-77 RADIO_SET

NUMBER OF YEARS (N) TO REPLACEMENT	SAVINGS
0	1410
1	1056
2	742
3	464
4	218

NOTE: SAVINGS ARE PER UNIT OVER REMAINING ECONOMIC
LIFE OF CURRENT TMS (5 YEARS)
ECONOMIC LIFE OF CURRENT TMS = 5
ECONOMIC LIFE OF REPLACEMENT TMS = 10
PERCENTAGE REDUCTION IN O&S COSTS = 40%
DISCOUNT RATE = 10%
ACQUISITION COST = 1000

Table 10: WHENBUY Table for the Modified Item.

IV. PROGRAM LOGIC

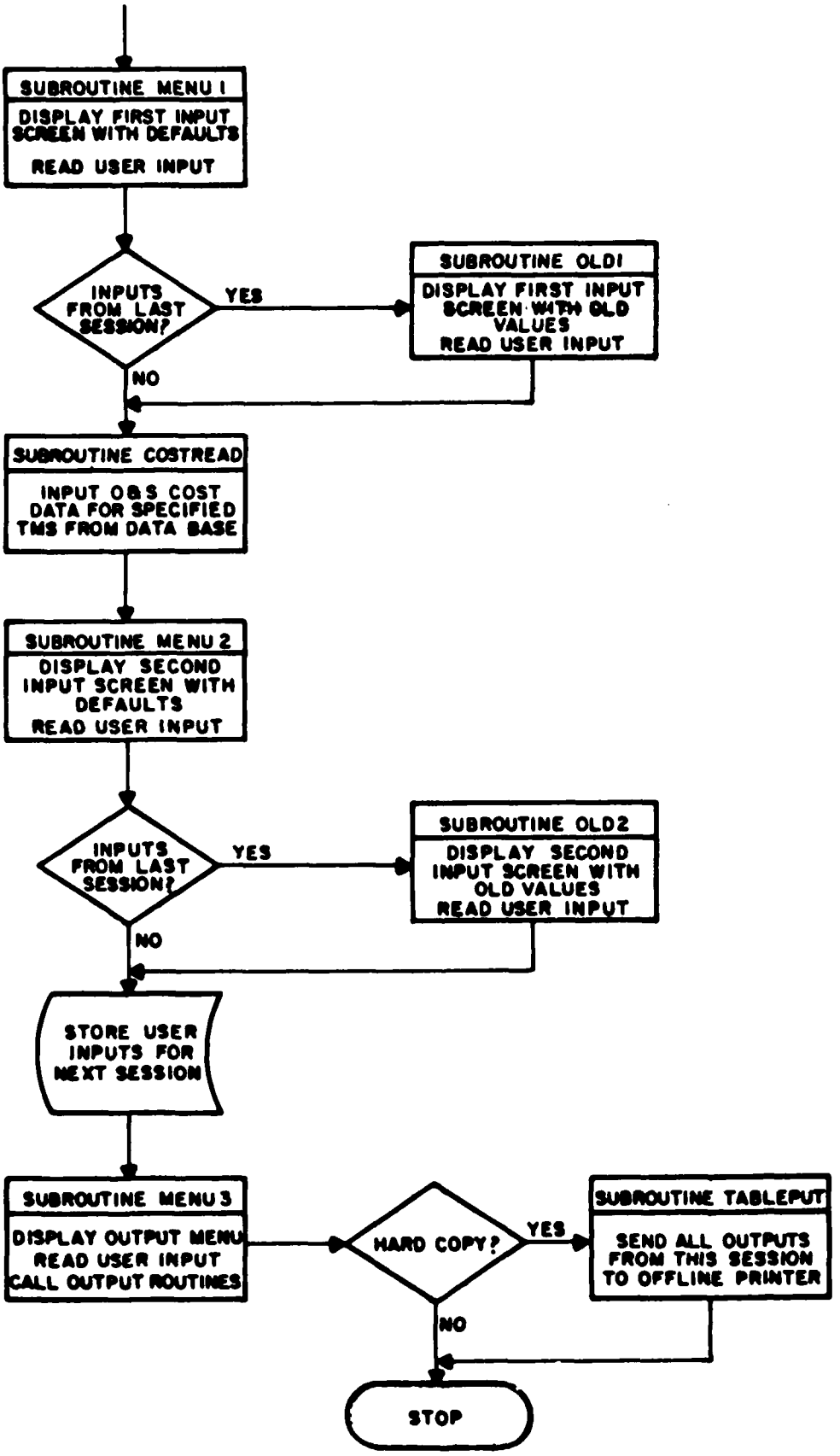
The interactive portion of COSTCASTER is a modular computer program which allows the user to obtain cost predictions and perform trade-off analyses for C-E end items. The main program serves primarily to organize the COSTCASTER session and call subroutines to display the input screens and output menu. This menu displays a list of available outputs and calls various subroutines to produce those outputs.

A brief description of each program module is given on the following pages. This description provides information about the purpose of each module and lists other modules which call and are called by it. In addition, flowcharts are given for the main program and the output menu subroutine. A complete listing of the program is given in the appendix.

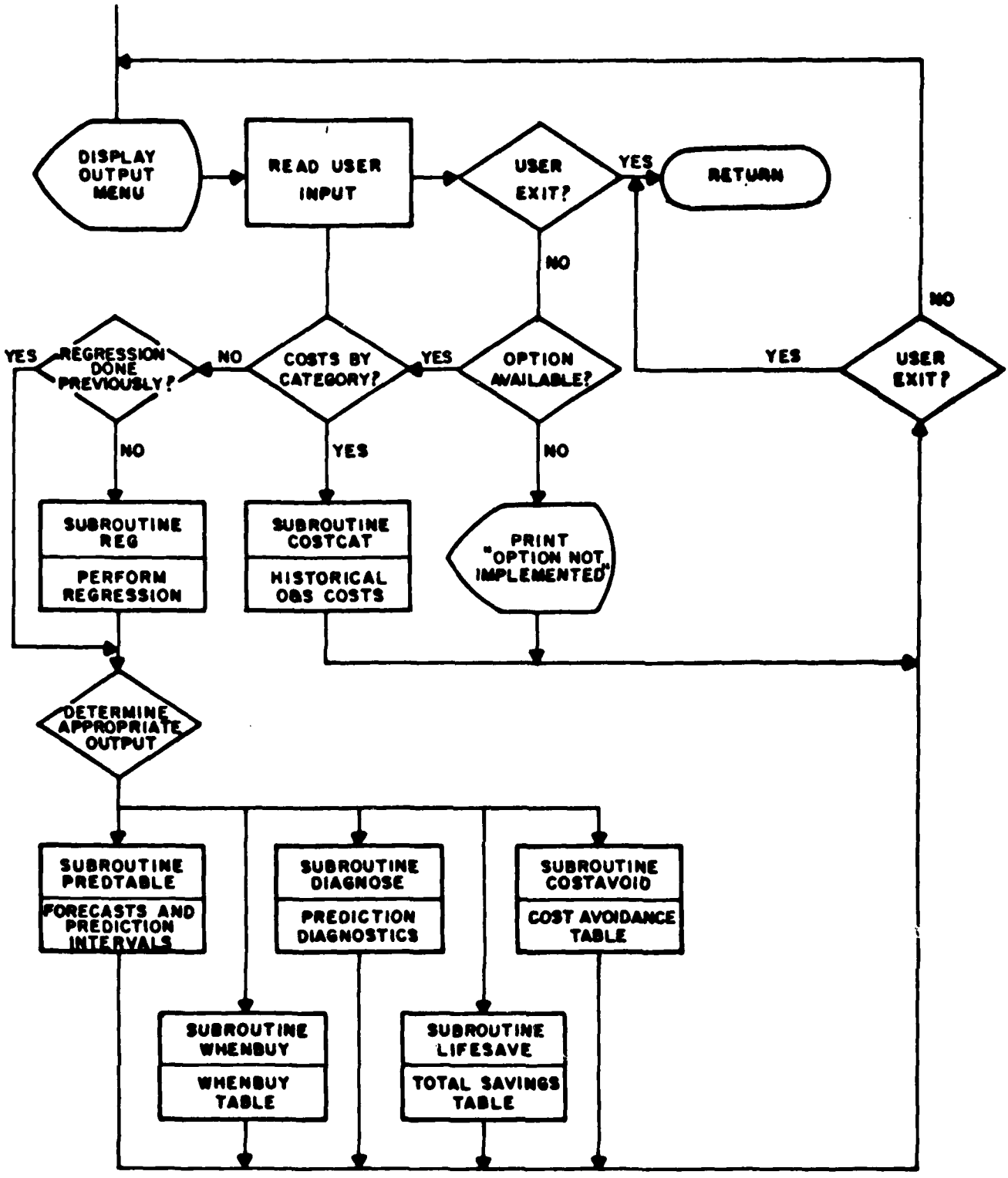
PROGRAM MODULE	PURPOSE	CALLS	CALLED BY
Main Program	Serves as Executive Program for COSTCASTER Session. Calls Menu and Printing Sub-routines.	COSTREAD MENU1 MENU2 MENU3 OLD1 OLD2 TABLEPUT	---
COSTAVOID	Produces the Cost Avoidance Tables.	PVOS	MENU3
COSTCAT	Produces the Table of Historical O&S Costs by Cost Category.	---	MENU3
COSTREAD	Reads the Historical Cost Data File From the Data Base.	---	Main Program
DIAGNOSE	Produces the Prediction Diagnostics Table.	REG	MENU3
LIFESAVE	Produces the Total Savings Table.	PVOS	MENU3
MENU1	Displays the First Input Screen and Reads User Input.	---	Main Program OLD1
MENU2	Displays the Second Input Screen and Reads User Input.	---	Main Program OLD2
MENU3	Displays the Output Products and Calls Specified Subroutines.	COSTAVOID COSTCAT DIAGNOSE LIFESAVE PREDTABLE REG WHENBUY	Main Program

PROGRAM MODULE	PURPOSE	CALLS	CALLED BY
OLD1	Recalls User Inputs From Last Session. Calls MENU1 to Redisplay First Input Screen.	MENU1	Main Program
OLD2	Recalls User Inputs From Last Session. Calls MENU2 to Redisplay Second Input Screen.	MENU2	Main Program
PREDTABLE	Produces O&S Cost Forecasts, Prediction Intervals, and Refined BERP Value.	PVOS	MENU3
PVOS	Calculates the Present Value of Predicted O&S Costs.	---	COSTAVOID LIFESAVE PREDTABLE WHENBUY
REG	Performs Log-Log Weighted Least Squares Regression	TOTCOST	DIAGNOSE MENU3
TABLEPUT	Sends Outputs From Session to Printer.	---	Main Program
TOTCOST	Calculates the Sum of the O&S Costs Over the User-Specified Categories.	---	REG
WHENBUY	Produces WHENBUY Table.	PVOS WHERE	MENU3
WHERE	Calculates the Expected Overall Reduction in O&S Costs, Given User-Specified Reductions for Individual Cost Categories.	---	WHENBUY

COSTCASTER MAIN PROGRAM



SUBROUTINE MENU3



V. REFERENCES

[1] Burns, K.C., G.J. Zunic, D.E. Smith, J.G. Levine, and R.L. Gardner, "Methodology Underlying COSTCASTER, A Cost-Prediction And Trade-Off Model for Air Force Ground C-E Equipment," Technical Report No. 118-4, Desmatics, Inc., 1984.

[2] Draper, N.R., and H. Smith, Applied Regression Analysis, 2nd ed., John Wiley & Sons, New York, 1981.

[3] U.S. Air Force, Air Force Regulation 400-63, Volume II, "Communications-Electronics Logistics Support Cost Management Program," 1979.

APPENDIX: SOURCE LISTING

This appendix contains a source listing for the interactive portion of COSTCASTER. This program is currently running on an IBM 4381 mainframe computer. The program is written in REXX, a high level programming language available on the CMS system. It is designed to run interactively.

```
/*     COSTCASTER MENU     */
TRACE OFF
NUMERIC DIGITS 9
/* INTIALIZE VARIABLES */
DATAPOINT = 100
DATA.7.30 = NO
DATA.9.40 ='AN/FCC-00'
DATA.11.45= 10
DATA.13.60= '?'
DATA.16.60= 1
DATA.18.60= 1
DATA.20.60 ='EQUAL'
DATA.9.68 = NO
DATA.13.25 = Y
DATA.14.25 = Y
DATA.15.25 = Y
DATA.16.25 = Y
DATA.17.25 = Y
DATA.18.25 = Y
DATA.19.25 = Y
DATA.20.25 = Y
DATA.21.25 = Y
DATA.22.25 = Y
DATA.13.65 = Y
DATA.14.65 = Y
DATA.15.65 = Y
DATA.16.65 = Y
DATA.17.65 = Y
DATA.18.65 = Y
DATA.19.65 = Y
DATA.20.65 = Y
DATA.21.65 = Y
DATA.22.65 = Y
DATA.13.31 = 50
DATA.14.31 = 50
DATA.15.31 = 50
DATA.16.31 = 50
DATA.17.31 = 50
DATA.18.31 = 50
DATA.19.31 = 50
DATA.20.31 = 50
DATA.21.31 = 50
DATA.22.31 = 50
DATA.13.71 = 50
DATA.14.71 = 50
DATA.15.71 = 50
DATA.16.71 = 50
DATA.17.71 = 50
DATA.18.71 = 50
DATA.19.71 = 50
DATA.20.71 = 50
DATA.21.71 = 50
DATA.22.71 = 50
```

```
/* USE OLD INPUT?     */
/*   TMS              */
/* DISCOUNT RATE      */
 /* ACQUISITION COST  */
/* CURRENT ECON LIFE  */
/* MOD/REP ECON LIF   */
/* USE WEIGHTS?       */
/* USE OLD INPUT      */
/* COST 1             */
/* COST 2             */
/* COST 3             */
/* COST 4             */
/* COST 5             */
/* COST 6             */
/* COST 7             */
/* COST 8             */
/* COST 9             */
/* COST 10            */
/* COST 11            */
/* COST 12            */
/* COST 13            */
/* COST 14            */
/* COST 15            */
/* COST 16            */
/* COST 17            */
/* COST 18            */
/* COST 19            */
/* PHANTOM COST       */
/* COST 1 REDUCTION   */
/* COST 2 REDUCTION   */
/* COST 3 REDUCTION   */
/* COST 4 REDUCTION   */
/* COST 5 REDUCTION   */
/* COST 6 REDUCTION   */
/* COST 7 REDUCTION   */
/* COST 8 REDUCTION   */
/* COST 9 REDUCTION   */
/* COST10 REDUCTION   */
/* COST11 REDUCTION   */
/* COST12 REDUCTION   */
/* COST13 REDUCTION   */
/* COST14 REDUCTION   */
/* COST15 REDUCTION   */
/* COST16 REDUCTION   */
/* COST17 REDUCTION   */
/* COST18 REDUCTION   */
/* COST19 REDUCTION   */
/* PHANTOM REDUCTION   */
```

```
 PREDFLAG=0
 WHENFLAG=0
 COSTFLAG=0
 LIFEFLAG=0
 DIAGFLAG=0
 COSTCATFLAG=0
/* DISPLAY FIRST MENU AND READ IN USER CHANGES                          */
CURSOR = '7 29';CALL MENU1
/*  IF USER WANTS LAST SESSIONS INPUT, SHOW FIRST MENU WITH OLD INPUTS*/
IF DATA.7.30 = YES THEN CALL OLD1
/* DISPLAY SECOND MENU AND READ IN USER CHANGES                         */
CURSOR = '9 57';CALL MENU2
/*  IF USER WANTS LAST SESSIONS INPUT,SHOW SECOND MENU WITH OLD INPUTS*/
IF DATA.9.58 = YES THEN CALL OLD2  /*IF USER WANTS OLD INPUTS,DISPLAY THEM */
IF ACQCOST= ? THEN ACQCOST = -99  /* IF ACQ. COST NOT SPECIFIED,SET EQUAL TO
-99 */
/* DISPLAY THIRD MENU AND READ IN USER CHANGES                          */
CURSOR = '1 1';CALL MENU3
RESTART:  /* GO HERE IF USER WANTS TO QUIT */
/* OUTPUT THE USER INPUTS TO A FILE CALLED COST DATA A                  */
EXECIO 1 DISKW COST DATA A 1 '(STRING' TMS DISCRATE ACQCOST,
       OLDLIFE NEWLIFE DATA.20.60 COST.1 COST.2 COST.3 COST.4,
       COST.5 COST.6 COST.7 COST.8 COST.9 COST.10 COST.11,
       COST.12 COST.13 COST.14 COST.15 COST.16 COST.17 COST.18 COST.19
EXECIO 1 DISKW COST DATA A 2 '(STRING' COSTRED.1 COSTRED.2 COSTRED.3,
       COSTRED.4 COSTRED.5 COSTRED.6 COSTRED.7 COSTRED.8 COSTRED.9,
       COSTRED.10 COSTRED.11 COSTRED.12 COSTRED.13 COSTRED.14 COSTRED.15,
       COSTRED.16 COSTRED.17 COSTRED.18 COSTRED.19
/* ASK USER IF THEY WOULD LIKE HARD COPY */
SAY 'WOULD YOU LIKE A PRINTED COPY OF THE TABLES? (YES OR NO)'
PULL ANSWER
IF ANSWER=YES THEN CALL TABLEPUT /* IF YES, PRINT TABLES */
/* IF TABLE FILES WERE PRODUCED, ERASE THEM */
IF PREDFLAG=1 THEN ERASE PREDTABLE TABLE
IF WHENFLAG=1 THEN ERASE WHENBUY TABLE
IF COSTFLAG=1 THEN ERASE COSTAVOID TABLE
IF LIFEFLAG=1 THEN ERASE LIFESAVE TABLE
IF DIAGFLAG=1 THEN ERASE DIAGNOSE TABLE
IF COSTCATFLAG=1 THEN ERASE COSTCAT TABLE
EXIT
```

```
/*************************************************************************/
/* MENU1:                                                                */
/* SUBROUTINE TO DISPLAY FIRST MENU                                      */
/*                                                                       */
/*************************************************************************/
MENU1:
/* PUT MENU1 IN THE QUEUE */
QUEUE CURSOR UL IC
QUEUE 1 31 PH '''*** COSTCASTER ***'''
QUEUE 3 10 PL '''PLEASE ENTER THE INPUTS YOU WANT COSTCASTER TO USE FOR'''
QUEUE 4 10 PL '''THIS ANALYSIS AND PRESS RETURN'''
QUEUE 6 10 PL '''WOULD YOU LIKE COSTCASTER TO USE THE SAME INPUTS'''
QUEUE 7 10 PL '''AS LAST SESSION? '''
QUEUE 7 29 UH ''''||OVERLAY(DATA.7.30,'    ')||'''' PL
QUEUE 9 1  PL '''WHAT TMS WOULD YOU LIKE TO ANALYZE? '''
QUEUE 9 39 UH ''''||OVERLAY(DATA.9.40,'              ')||''''PL
QUEUE 11 1 PL '''WHAT DISCOUNT RATE WOULD YOU LIKE TO USE? '''
QUEUE 11 44 UH ''''||OVERLAY(DATA.11.45,'  ')||'''' PL
QUEUE 11 47 PL '''%'''
QUEUE 13  1 PL '''WHAT IS THE ACQUISITION COST OF THE MOD/REPLACEMENT TMS?'''
QUEUE 13 59 UH ''''||OVERLAY(DATA.13.60,'          ') '''' PL
QUEUE 13 68 PL '''DOLLARS'''
QUEUE 14  1 PL '''(IF NOT SPECIFIED, UNIT PRICE OF CURRENT TMS USED)'''
QUEUE 16 1  PL '''WHAT IS THE REMAINING ECONOMIC LIFE OF THE CURRENT TMS? '''
QUEUE 16 59 UH ''''||OVERLAY(DATA.16.60,'  ')''''
QUEUE 16 63 PL '''YEARS'''
QUEUE 18 1  PL '''WHAT IS THE ECONOMIC LIFE OF THE MOD/REPLACEMENT TMS? '''
QUEUE 18 59 UH ''''||OVERLAY(DATA.18.60,'  ')||'''' PL
QUEUE 18 63 PL '''YEARS'''
QUEUE 20 1 PL  '''TYPE OF WEIGHTING IN THE PREDICTION SUBMODEL ?'''
QUEUE 20 59 UH ''''||OVERLAY(DATA.20.60,'      ')'''' PL
QUEUE 21 10 PL '''(EQUAL OR LINEAR)'''
QUEUE
'DVHMEN (STACK'   /* DVHMEN IS A UTILITY TO DISPLAY FULLSCREEN MENUS */
/*  READ IN ANY USER CHANGES                                             */
PULL . . . KEY
DO QUEUED()
   PULL . K L HOLD
   DATA.K.L=STRIP(HOLD)
END
CALL ERRORCHECK1   /* CHECK FOR INVALID INPUTS */
RETURN
```

```
/*********************************************************************/
/* MENU2:                                                            */
/* SUBROUTINE TO DISPLAY SECOND MENU                                 */
/*********************************************************************/
MENU2:
/* PUT MENU2 IN THE QUEUE */
QUEUE CURSOR UL IC
QUEUE 1 27 PH '''*** COSTCASTER ***'''
QUEUE 3 1 PL '''PLEASE INDICATE BELOW THOSE COST CATEGORIES WHICH YOU WOULD
LIKE TO USE FOR'''
QUEUE 4 1 PL '''THIS ANALYSIS BY PUTTING A Y(YES) OR N(NO) NEXT TO THE
APPROPRIATE CATEGORY'''
QUEUE 5 1 PL '''YOU MAY ALSO INDICATE THE EXPECTED COST REDUCTION
ASSOCIATED'''
QUEUE 6 1 PL ''' WITH THE MOD/REPLACEMENT TMS'''
QUEUE 9 1 PL '''DO YOU WANT TO USE THE SAME CATEGORIES AS LAST SESSION?'''
QUEUE 9 67 UH ''''||OVERLAY(DATA.9.68,'    ')'''' PL
QUEUE 11 6 PL '''COST CATEGORY          % REDUCTION      COST CATEGORY
% REDUCTION'''
QUEUE 13 1  PL '''OPERATIONS PERSONNEL'''
QUEUE 13 24 UH ''''||OVERLAY(DATA.13.25,' ')'''' PL
QUEUE 13 30 UH ''''OVERLAY(DATA.13.31,'    ')''''  PL
QUEUE 14 1  PL '''BASE MAINTENANCE PERS.    '''
QUEUE 14 24 UH ''''||OVERLAY(DATA.14.25,' ')'''' PL
QUEUE 14 30 UH ''''OVERLAY(DATA.14.31,'    ')''''  PL
QUEUE 15 1  PL '''ADMINISTRATIVE PERS.      '''
QUEUE 15 24 UH ''''||OVERLAY(DATA.15.25,' ')'''' PL
QUEUE 15 30 UH ''''OVERLAY(DATA.15.31,'    ')''''  PL
QUEUE 16 1  PL '''SUPPLY SUPPORT PERS.      '''
QUEUE 16 24 UH ''''||OVERLAY(DATA.16.25,' ')'''' PL
QUEUE 16 30 UH ''''OVERLAY(DATA.16.31,'    ')''''  PL
QUEUE 17 1  PL '''FUEL                      '''
QUEUE 17 24 UH ''''||OVERLAY(DATA.17.25,' ')'''' PL
QUEUE 17 30 UH ''''OVERLAY(DATA.17.31,'    ')''''  PL
QUEUE 18 1  PL '''MAINTENANCE MATERIEL      '''
QUEUE 18 24 UH ''''||OVERLAY(DATA.18.25,' ')'''' PL
QUEUE 18 30 UH ''''OVERLAY(DATA.18.31,'    ')''''  PL
QUEUE 19 1  PL '''UTILITIES                 '''
QUEUE 19 24 UH ''''||OVERLAY(DATA.19.25,' ')'''' PL
QUEUE 19 30 UH ''''OVERLAY(DATA.19.31,'    ')''''  PL
QUEUE 20 1  PL '''DEPOT MAINTENANCE         '''
QUEUE 20 24 UH ''''||OVERLAY(DATA.20.25,' ')'''' PL
QUEUE 20 30 UH ''''OVERLAY(DATA.20.31,'    ')''''  PL
QUEUE 21 1  PL '''REPLACEMENT INVESTMENT    '''
QUEUE 21 24 UH ''''||OVERLAY(DATA.21.25,' ')'''' PL
QUEUE 21 30 UH ''''OVERLAY(DATA.21.31,'    ')''''  PL
QUEUE 22 1  PL '''BASE OPERATING SUPPORT    '''
QUEUE 22 24 UH ''''||OVERLAY(DATA.22.25,' ')'''' PL
QUEUE 22 30 UH ''''OVERLAY(DATA.22.31,'    ')''''  PL
QUEUE 13 40 PL '''REAL PROPERTY MAINT.      '''
QUEUE 13 64 UH ''''||OVERLAY(DATA.13.65,' ')'''' PL
QUEUE 13 70 UH ''''OVERLAY(DATA.13.71,'    ')''''  PL
QUEUE 14 40 PL '''BASE COMMUNICATIONS       '''
QUEUE 14 64 UH ''''||OVERLAY(DATA.14.65,' ')'''' PL
QUEUE 14 70 UH ''''OVERLAY(DATA.14.71,'    ')''''  PL
```

```
QUEUE 16 40 P  '''MEDICAL(HEALTH CARE)      '''
QUEUE 16 64 UH  ''''||OVERLAY(DATA.16.66,' ')'''' PL
QUEUE 16 70 UH ''''OVERLAY(DATA.16.71,'    ')''''  PL
QUEUE 16 40 PL '''TDY                      '''
QUEUE 16 64 UH ''''||OVERLAY(DATA.16.66,' ')'''' PL
QUEUE 16 70 UH ''''OVERLAY(DATA.16.71,'    ')''''  PL
QUEUE 17 40 PL '''PCS                      '''
QUEUE 17 64 UH ''''||OVERLAY(DATA.17.66,' ')'''' PL
QUEUE 17 70 UH ''''OVERLAY(DATA.17.71,'    ')''''  PL
QUEUE 18 40 PL '''GENERAL DEPOT SUPPORT    '''
QUEUE 18 64 UH ''''||OVERLAY(DATA.18.66,' ')'''' PL
QUEUE 18 70 UH ''''OVERLAY(DATA.18.71,'    ')''''  PL
QUEUE 19 40 PL '''ENGINEERING SUPPORT     '''
QUEUE 19 64 UH ''''||OVERLAY(DATA.19.66,' ')'''' PL
QUEUE 19 70 UH ''''OVERLAY(DATA.19.71,'    ')''''  PL
QUEUE 20 40 PL '''TRANS. AND PACKAGING    '''
QUEUE 20 64 UH ''''||OVERLAY(DATA.20.66,' ')'''' PL
QUEUE 20 70 UH ''''OVERLAY(DATA.20.71,'    ')''''  PL
QUEUE 21 40 PL '''ADVANCED TRAINING       '''
QUEUE 21 64 UH ''''||OVERLAY(DATA.21.66,' ')'''' PL
QUEUE 21 70 UH ''''OVERLAY(DATA.21.71,'    ')''''  PL
QUEUE
/* READ IN ANY USER CHANGES */
'DVHMEN (STACK'
PULL . . . KEY
DO QUEUED()
   PULL . K L HOLD
   DATA.K.L=STRIP(HOLD)
END
CALL ERRORCHECK2       /* CHECK FOR INVALID INPUTS */
RETURN
```

```
/*************************************************************************/
/* MENU3:                                                                */
/* SUBROUTINE TO DISPLAY THIRD MENU                                      */
/*************************************************************************/
MENU3:
/* PUT MENU3 IN THE QUEUE */
QUEUE CURSOR UL IC
QUEUE 1 24 PH '''***** COSTCASTER *****'''
QUEUE 3 1 PL '''WHICH OF THE FOLLOWING COSTCASTER OUTPUTS WOULD YOU LIKE TO
SEE ? '''
QUEUE 6 2 PL '''PF1 - DATA INPUT TABLE'''
QUEUE 7 2 PL '''PF2 - DATA INPUT TABLE LISTED BY COST CATEGORY'''
QUEUE 9 2 PL '''PF3 - TABLE OF COST PREDICTIONS AND ASSOCIATED PREDICTION
INTERVALS'''
QUEUE 11 2 PL '''PF4 - TABLE OF PREDICTION DIAGNOSTICS'''
QUEUE 13 2 PL  '''PF5 - COST AVOIDANCE TABLES'''
QUEUE 15 2 PL '''PF6 - WHENBUY TABLES'''
QUEUE 17 2 PL '''PF7 - MAINTENANCE TABLES'''
QUEUE 19 2 PL '''PF8 - SAVINGS TABLE'''
QUEUE 21 2 PL '''PF9 - EXIT              '''
QUEUE
'DVHMEN (STACK'
PULL . . . KEY
/* DO WHAT THE USER ASKED */
IF KEY=PF10 THEN DO
         QUEUE '.ALARM'
         QUEUE 24 12 PH '''***PF10 NOT DEFINED       ***    '''
         CURSOR = ' 1 1 '
         SIGNAL MENU3
      END
IF KEY=PF09 THEN SIGNAL RESTART
IF KEY=PF01 THEN DO
         QUEUE '.ALARM'
         QUEUE 24 12 PH '''***OPTION NOT IMPLEMENTED***    '''
         CURSOR = ' 5 6 '
         SIGNAL MENU3
      END
IF KEY=PF02 THEN SIGNAL COSTCAT
IF KEY=PF12 THEN DO
         QUEUE '.ALARM'
         QUEUE 24 12 PH '''***PF KEY NOT DEFINED***    '''
         CURSOR = '1 1 '
         SIGNAL MENU3
      END
IF KEY=PF07 THEN DO
         QUEUE '.ALARM'
         QUEUE 24 12 PH '''***OPTION NOT IMPLEMENTED***    '''
         CURSOR = '17 2 '
         SIGNAL MENU3
      END
CALL REG
REGFLAG=1
IF KEY=PF03 THEN SIGNAL PREDTABLE
IF KEY=PF04 THEN SIGNAL DIAGNOSE
IF KEY=PF05 THEN SIGNAL COSTAVOID
IF KEY=PF06 THEN SIGNAL WHENBUY
IF KEY=PF08 THEN SIGNAL LIFESAVE
RETURN
```

```
/*********************************************************************/
/* OLD1:                                                             */
/* SUBROUTINE TO READ IN LAST SESSIONS DATA AND REDISPLAY MENU1      */
/*                                                                   */
/*********************************************************************/
OLD1:
EXECIO 1 DISKR COST DATA A 1
PULL DATA.9.40 DATA.11.45 DATA.13.60 DATA.16.60 DATA.18.60 DATA.20.60 .
IF DATA.13.60 = -99 THEN DATA.13.60 = ?
CURSOR = '7 29';CALL MENU1
RETURN
```

```
/*********************************************************************/
/* OLD2:                                                             */
/* SUBROUTINE TO READ IN LAST SESSIONS DATA AND REDISPLAY MENU2      */
/*                                                                   */
/*********************************************************************/
OLD2:
EXECIO 2 DISKR COST DATA A 1
PULL DATA.9.40 DATA.11.45 DATA.13.60 DATA.16.60 DATA.18.60 DATA.20.60,
     DATA.13.25 DATA.14.25 DATA.15.25 DATA.16.25,
     DATA.17.25 DATA.18.25 DATA.19.25 DATA.20.25,
     DATA.21.25 DATA.22.25 DATA.13.65 DATA.14.65,
     DATA.15.65 DATA.16.65 DATA.17.65 DATA.18.65,
     DATA.19.65 DATA.20.65 DATA.21.65
PULL DATA.13.31 DATA.14.31 DATA.15.31,
     DATA.16.31 DATA.17.31 DATA.18.31 DATA.19.31 DATA.20.31 DATA.21.31,
     DATA.22.31 DATA.13.71 DATA.14.71 DATA.15.71 DATA.16.71 DATA.17.71,
     DATA.18.71 DATA.19.71 DATA.20.71 DATA.21.71
/* CONVERT COST CATEGORY INDICATORS BACK TO Y OR N                   */
DO   I= 13 TO 22
   IF DATA.I.25 = 1 THEN DATA.I.25=Y
      ELSE DATA.I.25 = N
   IF DATA.I.65 = 1 THEN DATA.I.65 = Y
      ELSE DATA.I.65 = N
END
CURSOR = '9 67';CALL MENU2
RETURN
```

```
/*************************************************************************/
/*      ERRORCHECK1: SUBROUTINE TO CHECK FOR INPUT ERRORS ON MENU1   */
/*************************************************************************/
ERRORCHECK1:
USE1=DATA.7.30
TMS=DATA.9.40
DISCRATE=DATA.11.45
ACQCOST=DATA.13.60
OLDLIFE=DATA.16.60
NEWLIFE=DATA.18.60
WEIGHT=DATA.20.60
IF USE1  = YES & USE1  =NO
      THEN DO
         QUEUE '.ALARM'
         QUEUE 2 12 PH '''***RESPONSE MUST BE YES OR NO***'''
         CURSOR = ' 7 29'
         SIGNAL MENU1
      END
IF DISCRATE < 1
      THEN DO
        QUEUE '.ALARM'
        QUEUE 2 12 PH '''***DISCOUNT RATE MUST BE BETWEEN 1 AND 100***'''
        CURSOR = ' 11 44'
        SIGNAL MENU1
      END
IF ACQCOST < 0 & ACQCOST  = ?
      THEN DO
        QUEUE '.ALARM'
        QUEUE 2 12 PH '''***INVALID ACQUISITION COST ***'''
        CURSOR = ' 13 69'
        SIGNAL MENU1
      END
IF OLDLIFE < 0 | OLDLIFE >16
      THEN DO
        QUEUE '.ALARM'
        QUEUE 2 12 PH '''***ECONOMIC LIFE MUST BE BETWEEN 0 AND 16***'''
        CURSOR = ' 16 69'
        SIGNAL MENU1
      END
IF NEWLIFE < 0 | NEWLIFE >16
      THEN DO
        QUEUE '.ALARM'
        QUEUE 2 12 PH '''***ECONOMIC LIFE MUST BE BETWEEN 0 AND 16***'''
        CURSOR = ' 18 69'
        SIGNAL MENU1
      END
IF NEWLIFE < OLDLIFE
      THEN DO
        QUEUE '.ALARM'
        QUEUE 2 1 PH '''***ECONOMIC LIFE OF NEW TMS MUST BE GREATER THAN
ECONOMIC LIFE OF OLD TMS***'''
        CURSOR = ' 18 69'
        SIGNAL MENU1
      END
```

```
IF DATA.20.60  ='EQUAL'& DATA.20.60  ='LINEAR'
      THEN DO
         QUEUE '.ALARM'
         QUEUE 2 12 PH '''***INCORRECT RESPONSE***'''
         CURSOR = '20 72'
         SIGNAL MENU1
      END
RETURN




/*************************************************************************/
/*      ERRORCHECK2: SUBROUTINE TO CHECK FOR INPUT ERRORS ON MENU2       */
/*                                                                       */
/*************************************************************************/
ERRORCHECK2:
/* CONVERT COST CATEGORY INDICATORS BACK TO Y OR N                       */
DO   I= 13 TO 22
   IF DATA.I.25 = 1 THEN DATA.I.25=Y
   IF DATA.I.25 = 0 THEN DATA.I.25='N'
   IF DATA.I.65 = 1 THEN DATA.I.65 = Y
   IF DATA.I.65 = 0 THEN DATA.I.65 = 'N'
END
USE2=DATA.9.58
IF  DATA.9.58  = YES & DATA.9.58  =NO
      THEN DO
         QUEUE '.ALARM'
         QUEUE 8 20 PH '''***RESPONSE MUST BE YES OR NO***'''
         CURSOR = ' 9 67'
         SIGNAL MENU2
      END
     DO I = 13 TO 22
          IF DATA.I.25  =Y & DATA.I.25  ='N'
          THEN DO
            CURSOR = I 24
            QUEUE '.ALARM'
            QUEUE 8 20 PH '''***ONLY Y OR N ARE VALID RESPONSES***'''
            SIGNAL MENU2
          END
          IF DATA.I.65  = Y & DATA.I.65  ='N'
            THEN DO
            CURSOR = I 64
            QUEUE '.ALARM'
            QUEUE 8 20 PH '''***ONLY Y OR N ARE VALID RESPONSES***'''
            SIGNAL MENU2
          END
```

```
            IF ABS(DATA.I.31) < 1 & DATA.I.31  = 0
              THEN DO
              CURSOR = I 30
              QUEUE '.ALARM'
              QUEUE 7 20 PH '''*** % REDUCTION MUST BE AN INTEGER ***'''
              SIGNAL MENU2
            END
            IF ABS(DATA.I.71) < 1  & DATA.I.71  = 0
              THEN DO
              CURSOR = I 70
              QUEUE '.ALARM'
              QUEUE 7 20 PH '''*** % REDUCTION MUST BE AN INTEGER ***'''
              SIGNAL MENU2
            END
   END
/* SET COST INDICATORS TO ZERO OR ONE                                    */
DO   I= 13 TO 22
  J=I-12
  K=I-2
   IF DATA.I.25 = Y THEN COST.J=1
      ELSE COST.J = 0
   IF DATA.I.65 = Y THEN COST.K = 1
      ELSE COST.K = 0
  COSTRED.J = DATA.I.31
  COSTRED.K = DATA.I.71
END
RETURN
```

```
/*************************************************************************/
/*  COSTREAD SUBROUTINE: READ THE COST DATA FILE                         */
/*************************************************************************/
COSTREAD:
TRACE OFF
J = Ø;LINENUM = Ø;FOUND = Ø
DO WHILE FOUND = Ø
   LINENUM=LINENUM+1
EXECIO 1 DISKR CCINPUT XEDIT A LINENUM
/* IF EOF AND TMS NOT FOUND, GIVE MESSAGE*/
   IF RC=2 THEN DO
    QUEUE '.ALARM'
    QUEUE 2 12 PH '''***TMS IS NOT CURRENTLY IN THE DATA BASE***'''
    CURSOR= ' 9 39 '
    SIGNAL MENU1
    END
   PULL FTMS .
   IF FTMS  = TMS THEN NOP /* IF TMS NOT A MATCH, KEEP LOOKING */
      ELSE DO 1            /* IF TMS A MATCH,READ THE VARIABLES*/
               T = TMS
               J = 1
               FOUND = 1
               EXECIO 1 DISKR CCINPUT XEDIT A LINENUM
                  PULL FTMS SRD.J NOMEN.J NSN.J AVG.J YEAR.J,
                       PRICE.J COST.1.J COST.2.J COST.3.J,
                       COST.4.J COST.5.J COST.6.J COST.7.J,
                       COST.8.J COST.9.J COST.1Ø.J COST.11.J,
                       COST.12.J COST.13.J COST.14.J COST.15.J,
                       COST.16.J COST.17.J COST.18.J COST.19.J
               DO WHILE FTMS=TMS
                  T = TMS
                  J = J+1
                  EXECIO 1 DISKR CCINPUT XEDIT A
                  PULL FTMS SRD.J NOMEN.J NSN.J AVG.J YEAR.J,
                       PRICE.J COST.1.J COST.2.J COST.3.J,
                       COST.4.J COST.5.J COST.6.J COST.7.J,
                       COST.8.J COST.9.J COST.1Ø.J COST.11.J,
                       COST.12.J COST.13.J COST.14.J COST.15.J,
                       COST.16.J COST.17.J COST.18.J COST.19.J
               END
            END
END
N=J-1
IF ACQCOST=-99 THEN ACQCOST=PRICE.N /*SET ACQCOST=LASTBUY PRICE */
RETURN
```

```
/*********************************************************************/
/*   TOTCOST: SUBROUTINE TO CALCULATE THE SUM OF THE COSTS FOR       */
/*            THE USER SPECIFIED CATEGORIES                          */
/*********************************************************************/
TOTCOST:
   DO    I = 1 TO N
      TCOST.I = COST.1.I*COST.1 + COST.2.I*COST.2 + COST.3.I*COST.3,
           + COST.4.I*COST.4 + COST.5.I*COST.5 + COST.6.I*COST.6,
           + COST.7.I*COST.7 + COST.8.I*COST.8 + COST.9.I*COST.9,
           + COST.10.I*COST.10+ COST.11.I*COST.11+ COST.12.I*COST.12,
           + COST.13.I*COST.13+ COST.14.I*COST.14+ COST.15.I*COST.15,
           + COST.16.I*COST.16+ COST.17.I*COST.17+ COST.18.I*COST.18,
           + COST.19.I*COST.19
      IF TCOST.I = 0 THEN TCOST.I = .1 /* IF A COST IS ZERO,SET IS TO .1
DOLLARS */
      LTCOST.I = LOG(TCOST.I)
   END
RETURN




/*********************************************************************/
/* REG: A SUBROUTINE TO DO LOG-LOG REGRESSION                        */
/*********************************************************************/
REG:
CALL COSTREAD
CALL TOTCOST
XSUM=0;YSUM=0;XYSUM=0;XXSUM=0;YYSUM=0;DENOM=0;
IF DATAPOINT < N THEN INDEX = DATAPOINT /*DATAPOINT COMES FROM DIAGNOSE*/
 ELSE INDEX = N
    DO I=1 TO INDEX
       IF WEIGHT ='LINEAR' THEN WT = I
          ELSE WT = 1
       DENOM = DENOM + WT
       XSUM = XSUM + WT*INTLOG(I)
       YSUM = YSUM + WT*LTCOST.I
    END
YBAR = YSUM/DENOM
XBAR = XSUM/DENOM
    DO I=1 TO INDEX
       ILOG=INTLOG(I)
       IF WEIGHT = 'LINEAR' THEN WT = I
          ELSE WT = 1
      XXSUM =XXSUM+WT*(ILOG-XBAR)**2
      XYSUM =XYSUM + WT*(ILOG-XBAR)*(LTCOST.I-YBAR)
      YYSUM =YYSUM +WT*(LTCOST.I-YBAR)**2
   END
B1 =(XYSUM)/(XXSUM)
B0 = YBAR-B1*XBAR
MSE =(YYSUM-XYSUM**2/XXSUM)/(INDEX-2)
TRACE OFF
TRACE OFF
RETURN
```

```
/*************************************************************************/
/*   TEST EXP: THIS SUBROUTINE TAKES E AND RAISES IT TO THE POWER X      */
/*************************************************************************/
EXP:
ARG X
X = STRIP(X)
FLAG = 0
IF X < 0 THEN DO
    X = ABS(X)
    FLAG = 1
    END
EE= 1;FACT=1;TEMP=1
   DO ZZ = 1 TO 100
      TEMP=TEMP*(X/ZZ)
      EE = EE+TEMP
    IF ABS(TEMP) < 0.000000001 THEN DO
        IF FLAG = 1 THEN RETURN 1/EE
        ELSE RETURN EE
        END
   END
        IF FLAG = 1 THEN RETURN 1/EE
        ELSE RETURN EE
```

```
/*************************************************************************/
/*   TEST LOG: THIS SUBROUTINE CALCULATES THE NATURAL LOG OF X           */
/*************************************************************************/
LOG:
ARG VV
WW=0
VV = STRIP(VV)
IF VV = 0 THEN RETURN -230
CHECK=0
IF VV < 1 THEN DO WHILE CHECK < 1
         CHECK= VV*10**(WW+1)
         WW=WW+1
         E=-WW
         END
   ELSE E = LENGTH(TRUNC(VV))-1
M=VV/(10**E)
LN = 0;MANT = 0
   DO ZZ = 1 TO 100 BY 2
     SHEMP = 2*(1/ZZ)*((M-1)/(M+1))**ZZ
      MANT=MANT+SHEMP
   IF SHEMP < 0.000000001 THEN DO
         LN = MANT+E*2.3026861
         RETURN LN
      END
   END
LN = MANT+E*2.3026861
RETURN LN
```

```
/*************************************************************************/
/*   FSTAT:  THIS SUBROUTINE GIVES THE 95TH PERCENTILE OF THE F          */
/*           DISTRIBUTION FOR THE SPECIFIED DEGREES OF FREEDOM           */
/*************************************************************************/
FSTAT:
ARG DF
IF DF=1          THEN RETURN  161.4;IF DF=2          THEN RETURN   18.51
IF DF=3          THEN RETURN   10.13;IF DF=4          THEN RETURN   7.71
IF DF=5          THEN RETURN   6.61 ;IF DF=6          THEN RETURN   5.99
IF DF=7          THEN RETURN   5.59 ;IF DF=8          THEN RETURN   5.32
IF DF=9          THEN RETURN   5.12 ;IF DF=10         THEN RETURN   4.96
IF DF= 11        THEN RETURN   4.84 ;IF DF= 12        THEN RETURN   4.75
IF DF= 13        THEN RETURN   4.67 ;IF DF= 14        THEN RETURN   4.60
IF DF= 15        THEN RETURN   4.54 ;IF DF= 16        THEN RETURN   4.49
IF DF= 17        THEN RETURN   4.45 ;IF DF= 18        THEN RETURN   4.41
IF DF= 19        THEN RETURN   4.48 ;IF DF= 20        THEN RETURN   4.35
IF DF= 21        THEN RETURN   4.32 ;IF DF= 22        THEN RETURN   4.30
IF DF= 23        THEN RETURN   4.28 ;IF DF= 24        THEN RETURN   4.26
IF DF= 25        THEN RETURN   4.24 ;IF DF= 26        THEN RETURN   4.23
IF DF= 27        THEN RETURN   4.21 ;IF DF= 28        THEN RETURN   4.20
IF DF= 29        THEN RETURN   4.18 ;IF DF= 30        THEN RETURN   4.17
IF DF > 30       THEN RETURN   4.17
```

```
/*************************************************************************/
/*   PVOS: SUBROUTINE TO CALCULATE THE PRESENT VALUE OF O&S COSTS        */
/*         OF THE CURRENT TMS OVER ITS REMAINING ECONOMIC LIFE           */
/*************************************************************************/
PVOS:
TOTALOS=0
  DO K = START TO STOP
     PREDC= EXP(B0+B1*INTLOG(K+N))
     IF PREDC< 1 THEN PREDC= 0
     TOTALOS = TOTALOS +((1+DISCRATE/100)**-K)*PREDC
  END
RETURN
```

```
/*************************************************************************/
/* PREDTABL: SUBROUTINE TO DISPLAY PRED. O&S COST AND PRED. INTERVALS*/
/*        FOR THE CURRENT TMS OVER ITS REMAINING ECONOMIC LIFE      */
/*************************************************************************/
PREDTABLE:
IF PREDFLAG=1 THEN SIGNAL PREDTABLEBEGIN
PREDFLAG=1
EXECIO 1  DISKW PREDTABL TABLE A 1  '(STRING' '-LAB1 '
EXECIO 1 DISKW  PREDTABLE TABLE A 2          '(STRING' 1 5 '''',
CENTER('PREDICTIONS FOR THE' TMS NOMEN.N,64)''''
EXECIO 1 DISKW  PREDTABLE TABLE A 3          '(STRING' 2 5 '''',
CENTER('THE PREDICTED COST FOR YEAR T ='TRUNC(EXP(B0)+.5)'*T**('TRUNC(B1,4)')
(FY82 DOLLARS)',64)''''
EXECIO 1  DISKW PREDTABL TABLE A 4  '(STRING' 3 5 '  '
EXECIO 1 DISKW  PREDTABLE TABLE A 5          '(STRING' 4 5 '''',
 CENTER('FY',6) CENTER('T',6) CENTER('COST',9) CENTER('FORECAST',13),
    CENTER('95% PREDICTION INTERVAL',25)''''
EXECIO 1 DISKW  PREDTABLE TABLE A 6          '(STRING' 5 5 '''',
    CENTER('--',6) CENTER('-',6) CENTER('----',9) CENTER('--------',13),
    CENTER('-----------------------',25)''''
LINE=7;PLACE=6
    DO I = 1 TO N
EXECIO 1 DISKW  PREDTABLE TABLE A LINE       '(STRING' PLACE 5 '''',
        CENTER(YEAR.I,6) CENTER(I,6) CENTER(TRUNC(TCOST.I+.5),9),
        CENTER('*',13) CENTER('*',25)''''
        LINE=LINE+1;PLACE=PLACE+1
    END
DF = N-2
CURRF=FSTAT(DF)
YR = YEAR.N + 1
    DO I = N+1 TO OLDLIFE + N
       ILOG=INTLOG(I)
       PMEAN = B0 + B1*ILOG
       PMULT1= CURRF*MSE
       PMULT2=(1/WT)+(1/DENOM)+((ILOG-XBAR)**2)/XXSUM
       PMULT = EXP(0.5*LOG(PMULT1*PMULT2))
       LB = EXP(PMEAN-PMULT)
       UB = EXP(PMEAN+PMULT)
    IF PLACE= 23 THEN DO
       EXECIO 1 DISKW  PREDTABL TABLE A LINE '(STRING' 23 1 PH '''
                PRESS PF12 TO SEE A LIST OF OPTIONS. '''
       EXECIO 1 DISKW  PREDTABL TABLE A LINE+1 '(STRING' 24 1 PH '''
                PRESS PF1 TO SEE NEXT SCREEN,PF2 TO EXIT. '''
       EXECIO 1  DISKW PREDTABL TABLE A LINE+2' (STRING' '-LAB2 '
EXECIO 1 DISKW  PREDTABLE TABLE A LINE+3     '(STRING' 2 5 '''',
CENTER('THE PREDICTED COST FOR YEAR T ='TRUNC(EXP(B0)+.5)'*T**('TRUNC(B1,4)')
(FY82 DOLLARS)',64)''''
EXECIO 1 DISKW  PREDTABLE TABLE A LINE+4     '(STRING' 4 5 '''',
 CENTER('FY',6) CENTER('T',6) CENTER('COST',9) CENTER('FORECAST',13),
    CENTER('95% PREDICTION INTERVAL',25)''''
EXECIO 1 DISKW  PREDTABLE TABLE A LINE+5     '(STRING' 5 5 '''',
    CENTER('--',6) CENTER('-',6) CENTER('----',9) CENTER('--------',13),
    CENTER('-----------------------',25)''''
```

```
        PLACE=6
        LINE=LINE+6
        PREDTABLEFLAG=1
        END
EXECIO 1 DISKW  PREDTABLE TABLE A LINE         '(STRING' PLACE 6 '''',
         CENTER(YR,6) CENTER(I,6) CENTRE('*',9),
         CENTER(TRUNC(EXP(PMEAN)+.5),13),
         CENTER('('TRUNC(LB+.5)','TRUNC(UB+.5)')',25)''''
         YR = YR + 1
         LINE=LINE+1;PLACE=PLACE+1
     END
START=1;STOP=OLDLIFE
CALL PVOS
       EXECIO 1 DISKW  PREDTABL TABLE A LINE '(STRING' PLACE 6 '''' ''''
       EXECIO 1 DISKW  PREDTABL TABLE A LINE+1 '(STRING' PLACE+1 6 '''',
'BASED ON USER INPUTS,BERP=' TRUNC((ACQCOST/(TOTALOS+.5))+.005,2)  ''''
IF PREDTABLEFLAG=1 THEN DO
       EXECIO 1 DISKW  PREDTABL TABLE A LINE+2 '(STRING' 24 1 PH '''
                PRESS PF1 TO SEE PREVIOUS SCREEN,PF2 TO EXIT. '''
       END
ELSE   EXECIO 1 DISKW  PREDTABL TABLE A LINE+2 '(STRING' 24 7 PH,
            '''PRESS PF12 FOR A LIST OF OPTIONS,PF2 TO EXIT. '''
START=1;STOP=OLDLIFE
CALL PVOS
IF WEIGHT='LINEAR' THEN EXECIO 1 DISKW  PREDTABL TABLE A LINE+1 '(STRING'
PLACE+1 6 '''',
'BASED ON USER INPUTS,BERP=' TRUNC((ACQCOST/(TOTALOS+.5))+.005,2),
'LINEAR WEIGHTS ARE USED.' ''''
IF WEIGHT='EQUAL' THEN EXECIO 1 DISKW  PREDTABL TABLE A LINE+1 '(STRING'
PLACE+1 6 '''',
'BASED ON USER INPUTS,BERP=' TRUNC((ACQCOST/TOTALOS)+.005,2),
'EQUAL WEIGHTS ARE USED.' ''''
PREDTABLEBEGIN:
'DVHMEN PREDTABLE TABLE A -LAB1'
PULL . . . KEY
CALL FUNCTION
IF PREDTABLEFLAG=1 THEN DO
 'DVHMEN PREDTABLE TABLE A -LAB2'
PULL . . . KEY
IF KEY = PF01 THEN SIGNAL PREDTABLEBEGIN
  ELSE CALL FUNCTION
TRACE OFF
END
RETURN
```

```
/*************************************************************************/
/* WHERE:   SUBROUTINE TO CALCULATE THE EXPECTED REDUCTION               */
/*          IN O&S COSTS                                                 */
/*************************************************************************/
WHERE:
     SUMCOST= 0
   DO I = 1 TO 19
      SUM.I = 0
      DO J= 1 TO N
      IF COST.I = 1 THEN DO
      IF WEIGHT ='LINEAR' THEN WT =J/N
         ELSE WT = 1
         SUM.I = WT*COST.I.J+SUM.I
         END
      END
   SUMCOST=SUMCOST+SUM.I
   END
NUMER= 0
DO I =1 TO 19
  NUMER =NUMER + SUM.I*(COSTRED.I/100)
END
WHERE =NUMER/SUMCOST
RETURN




/*************************************************************************/
/* WHENBUY: SUBROUTINE TO PRODUCE THE  WHENBUY TABLE                     */
/*************************************************************************/
WHENBUY:
IF WHENFLAG=1 THEN SIGNAL WHENBUYBEGIN
WHENFLAG=1
EXECIO 1  DISKW WHENBUY TABLE A 1      '(STRING' '-LAB1'
EXECIO 1  DISKW WHENBUY TABLE A 2       '(STRING' 1 1 '''',
          CENTER('EXPECTED SAVINGS IF REPLACEMENT IS MADE N YEARS FROM
PRESENT',65)''''
EXECIO 1  DISKW WHENBUY TABLE A 3       '(STRING' 2 1 '''',
          CENTER(TMS NOMEN.N,60) ''''
EXECIO 1  DISKW WHENBUY TABLE A 4       '(STRING' 3 1 '''' ''''
EXECIO 1  DISKW WHENBUY TABLE A 6       '(STRING' 4        1 '''',
          CENTER('NUMBER OF YEARS (N) TO REPLACEMENT',40)
CENTER('SAVINGS',20) ''''
LINE=6;PLACE=6
CALL WHERE
   WFLAG = 0
 DO I = 0 TO OLDLIFE-1
    START = I+1
    STOP = OLDLIFE
    LINE=LINE+1
```

```
    PLACE=PLACE+1
    CALL PVOS
    SAV1=WHERE*TOTALOS-ACQCOST*(1+DISCRATE/100)**-I
    SAV2=((1+DISCRATE/100)**-OLDLIFE)*ACQCOST*((NEWLIFE-OLDLIFE+I)/NEWLIFE)
    SAVING=SAV1+SAV2
    IF PLACE= 23 THEN DO
       EXECIO 1 DISKW  PREDTABL TABLE A LINE '(STRING' 23 1 PH '''
                PRESS PF12 TO SEE A LIST OF OPTIONS. '''
       EXECIO 1 DISKW  WHENBUY TABLE A LINE+1 '(STRING' 24 1 PH '''
                PRESS PF1 TO SEE NEXT SCREEN,PF2 TO EXIT. '''
       EXECIO 1  DISKW WHENBUY TABLE A LINE+2  '(STRING' '-LAB2 '
       PLACE=1
       LINE=LINE+3
      WFLAG=1
       END
    EXECIO 1  DISKW WHENBUY TABLE A LINE  '(STRING' PLACE    1 '''',
              CENTER(I,40) CENTER(FORMAT(SAVING,8,0),20) ''''
 END
PLACE=PLACE+1
IF PLACE+7>22 &  WFLAG  = 1  THEN DO
       EXECIO 1 DISKW  WHENBUY TABLE A LINE+1 '(STRING' 24 7 PH,
'''PRESS PF1 TO SEE PREVIOUS SCREEN,PF12 FOR OTHER OPTIONS . '''
       EXECIO 1  DISKW WHENBUY TABLE A LINE+2 '(STRING' '-LAB2 '
       PLACE=3
       LINE=LINE+2
      WFLAG = 1
       END
EXECIO 1  DISKW WHENBUY TABLE A LINE+1 '(STRING' PLACE 1 ''' '''
EXECIO 1  DISKW WHENBUY TABLE A LINE+2 '(STRING' PLACE+1  1 ,
     ''' NOTE: SAVINGS ARE PER UNIT OVER REMAINING ECONOMIC '''
EXECIO 1  DISKW WHENBUY TABLE A LINE+3 '(STRING' PLACE+2  1 '''',
   '      LIFE OF CURRENT TMS ('OLDLIFE' YEARS)' ''''
EXECIO 1  DISKW WHENBUY TABLE A LINE+4 '(STRING' PLACE+3  1 '''',
          'ECONOMIC LIFE OF CURRENT TMS =' OLDLIFE ''''
EXECIO 1  DISKW WHENBUY TABLE A LINE+5 '(STRING' PLACE+4  1 '''',
          'ECONOMIC LIFE OF REPLACEMENT TMS = ' NEWLIFE '''
EXECIO 1  DISKW WHENBUY TABLE A LINE+6 '(STRING' PLACE+5  1 '''',
          'PERCENTAGE REDUCTION IN O&S COSTS = '
TRUNC(WHERE*100+.6*SIGN(WHERE))'%' ''''
EXECIO 1  DISKW WHENBUY TABLE A LINE+7 '(STRING' PLACE+6  1 '''',
          'DISCOUNT RATE =' DISCRATE'%' ''''
EXECIO 1  DISKW WHENBUY TABLE A LINE+8 '(STRING' PLACE+7  1 '''',
          'ACQUISITION COST =' ACQCOST ''''
IF WFLAG= 1 THEN EXECIO 1 DISKW WHENBUY TABLE A LINE+9 '(STRING' 24 1 PH,
             '''PRESS PF1 TO SEE PREVIOUS SCREEN,PF12 FOR OTHER OPTIONS . '''
ELSE EXECIO 1 DISKW  WHENBUY TABLE A LINE+9 '(STRING' 24 1 PH '''
               PRESS PF2 TO EXIT,PF12 FOR OTHER OPTIONS . '''
WHENBUYBEGIN:
'DVHMEN WHENBUY TABLE A  -LAB1'
PULL . . . KEY
IF KEY = PF02 THEN RETURN
CALL FUNCTION
IF WFLAG=1 THEN DO
'DVHMEN WHENBUY TABLE A  -LAB2'
PULL . . . KEY
IF KEY = PF01 THEN SIGNAL WHENBUYBEGIN
    END
CALL FUNCTION
RETURN
```

```
/*********************************************************************/
/* COSTAVOID: SUBROUTINE TO PRODUCE THE  COST AVOID TABLES           */
/*********************************************************************/
COSTAVOID:
IF COSTFLAG=1 THEN SIGNAL COSTAVOIDBEGIN
COSTFLAG=1
EXECIO 1 DISKW COSTAVOID TABLE A 1 '(STRING' '-LAB1'
EXECIO 1 DISKW COSTAVOID TABLE A 2 '(STRING'  1 1 '''',
  CENTER('EXPECTED COST AVOIDANCE AFTER N YEARS',60)''''
EXECIO 1 DISKW COSTAVOID TABLE A 3 '(STRING'  2 1 '''',
  CENTER('FOR THE' TMS NOMEN.N,60)''''
EXECIO 1 DISKW COSTAVOID TABLE A 4 '(STRING'  3 1 '''',
CENTER('DISCOUNT RATE='DISCRATE'%' 'ACQUISITION COST=$'ACQCOST,60)''''
EXECIO 1 DISKW COSTAVOID TABLE A 5 '(STRING'  4 1 '  '
EXECIO 1 DISKW COSTAVOID TABLE A 6 '(STRING'  5 5 '''',
    CENTER('YEARS AFTER PURCHASE',20) CENTER('PERCENT REDUCTION IN SUPPORT
COSTS',40)''''
EXECIO 1 DISKW COSTAVOID TABLE A 7 '(STRING'  6 5 '''',
    RIGHT(' ',19) RIGHT('10%',8) RIGHT('30%',8) RIGHT('50%',8)
RIGHT('70%',8),
RIGHT('90%',8)''''
EXECIO 1 DISKW COSTAVOID TABLE A 8 '(STRING'  7 5 '''',
    RIGHT(' ',19) RIGHT('---',8) RIGHT('---',8) RIGHT('---',8)
RIGHT('---',8),
RIGHT('---',8)''''
LINE=9;PLACE=8
  DO I = 1 TO OLDLIFE
     START=1
     STOP=I
     CALL PVOS
     DO J = 1 TO 9 BY 2
        COSTAVOID.J = (J/10)*TOTALOS-ACQCOST
     END
EXECIO 1 DISKW COSTAVOID TABLE A LINE '(STRING' PLACE 5 '''',
     CENTER(I,19) RIGHT(FORMAT(COSTAVOID.1,8,0),8),
      RIGHT(FORMAT(COSTAVOID.3,8,0),8) RIGHT(FORMAT(COSTAVOID.5,8,0),8),
      RIGHT(FORMAT(COSTAVOID.7,8,0),8) RIGHT(FORMAT(COSTAVOID.9,8,0),8)''''
LINE=LINE+1;PLACE=PLACE+1
  END
EXECIO 1 DISKW  COSTAVOID TABLE A LINE '(STRING' 24 1 PH '''
               PRESS PF2 TO EXIT,PF12 FOR OTHER OPTIONS . '''
COSTAVOIDBEGIN:
'DVHMEN COSTAVOID TABLE A  -LAB1'
PULL . . . KEY
IF KEY = PF02 THEN RETURN
CALL FUNCTION
  RETURN
```

```
/*********************************************************************/
/* LIFESAVE:SUBROUTINE TO PRODUCE THE                                */
/*          SAVINGS OVER ECONOMIC LIFE TABLE                         */
/*********************************************************************/
LIFESAVE:
IF LIFEFLAG=1 THEN SIGNAL LIFESAVEBEGIN
LIFEFLAG=1
EXECIO 1 DISKW LIFESAVE TABLE A 1 '(STRING' 1 1 '''',
    CENTER('TOTAL SAVINGS OVER THE LIFE OF ITEM REPLACING ',65)''''
EXECIO 1 DISKW LIFESAVE TABLE A 2 '(STRING' 2 1 '''',
    CENTER('THE' TMS NOMEN.N,65)''''
EXECIO 1 DISKW LIFESAVE TABLE A 3 '(STRING' 3 1 '''',
CENTER('DISCOUNT RATE='DISCRATE'%' 'ACQUISITION COST=$'ACQCOST,60)''''
EXECIO 1 DISKW LIFESAVE TABLE A 4 '(STRING' 4 1 ' '
EXECIO 1 DISKW LIFESAVE TABLE A 5 '(STRING' 5 1 '''',
    CENTER('LIFE OF REPLACEMENT',19) CENTER('PERCENT REDUCTION IN SUPPORT
COSTS',45)''''
EXECIO 1 DISKW LIFESAVE TABLE A 6 '(STRING' 6 1 '''',
    RIGHT(' ',19) RIGHT('10%',8) RIGHT('30%',8) RIGHT('50%',8)
RIGHT('70%',8),
RIGHT('90%',8)''''
EXECIO 1 DISKW LIFESAVE TABLE A 7 '(STRING' 7 1 '''',
    RIGHT(' ',19) RIGHT('---',8) RIGHT('---',8) RIGHT('---',8)
RIGHT('---',8),
RIGHT('---',8)''''
LINE=8;PLACE=8
  DO I = OLDLIFE TO 15
     START=1
     STOP=OLDLIFE
     CALL PVOS
     DO J = 1 TO 9 BY 2
        COSTAVOID.J = (J/10)*TOTALOS-ACQCOST,
        +ACQCOST*((1+DISCRATE/100)**-OLDLIFE)*(I-OLDLIFE)/I
     END
EXECIO 1 DISKW LIFESAVE TABLE A LINE '(STRING' PLACE 1 '''',
      CENTER(I,19) RIGHT(FORMAT(COSTAVOID.1,8,0),8),
      RIGHT(FORMAT(COSTAVOID.3,8,0),8) RIGHT(FORMAT(COSTAVOID.5,8,0),8),
      RIGHT(FORMAT(COSTAVOID.7,8,0),8) RIGHT(FORMAT(COSTAVOID.9,8,0),8)'''
LINE=LINE+1;PLACE=PLACE+1
  END
EXECIO 1 DISKW  LIFESAVE TABLE A LINE '(STRING' 24 7 PH '''
                PRESS PF2 TO EXIT,PF12 FOR OTHER OPTIONS . '''
LIFESAVEBEGIN:
'DVHMEN LIFESAVE TABLE A '
PULL . . . KEY
CALL FUNCTION
  RETURN
```

```
/*************************************************************************/
/* DIAGNOSE:SUBROUTINE TO PRODUCE THE PREDICTION                         */
/*          DIAGNOSTICS TABLE                                            */
/*************************************************************************/
DIAGNOSE:
IF DIAGFLAG=1 THEN SIGNAL DIAGNOSEBEGIN
DIAGFLAG=1
DO K = 3 TO N
   DATAPOINT=K
   CALL REG
   B0.K=B0
   B1.K=B1
   MSE.K=MSE
   XBAR.K=XBAR
   XXSUM.K=XXSUM
END
DO I = 4 TO N+1
   DO J = 3 TO N
        PMEAN = B0.J+B1.J*INTLOG(I)
        PRED.I.J=EXP(PMEAN)
   END
END
DO I = 4 TO N
   J = I-1
   K = I+1
   DF = J-2
        ILOG=INTLOG(I)
        IF WEIGHT ='LINEAR' THEN DO
        WT = I
        DENOM = J*(J+1)/2
        END
           ELSE WT = 1
        PMEAN = B0.J+B1.J*ILOG
        PMULT1= FSTAT(DF)*MSE.J
        PMULT2=(1/WT)+(1/DENOM)+((ILOG-XBAR.J)**2)/XXSUM.J
        PMULT = EXP(0.5*LOG(PMULT1*PMULT2))
        LB = EXP(PMEAN-PMULT)
        UB = EXP(PMEAN+PMULT)
        PI.I = 60*(UB-LB)/PRED.I.J
        RES.I = TCOST.I-PRED.I.J
        RAI.I = 100*RES.I/PRED.I.J
        SI.K = 100*(PRED.K.I-PRED.K.J)/PRED.K.J
END
LAST = N+1
   DF = N-2
        IF WEIGHT ='LINEAR' THEN DO
        WT = LAST
        DENOM = N*(N+1)/2
        END
        PMULT1= FSTAT(DF)*MSE.N
        PMULT2=(1/WT)+(1/DENOM)+((INTLOG(LAST)-XBAR.N)**2)/XXSUM.N
        PMULT = EXP(0.5*LOG(PMULT1*PMULT2))
        LB = EXP(PMEAN-PMULT)
        UB = EXP(PMEAN+PMULT)
PI.LAST = 60*(UB-LB)/PRED.LAST.N
```

```
EXECIO 1 DISKW DIAGNOSE TABLE A 1 '(STRING' 2 6 '''',
    CENTER('PREDICTION DIAGNOSTICS FOR THE' TMS NOMEN.N,60)''''
EXECIO 1 DISKW DIAGNOSE TABLE A 2 '(STRING' 3 6 '  '
EXECIO 1 DISKW DIAGNOSE TABLE A 3 '(STRING' 4 6 '''',
    CENTER('ACCURACY INDICES',60)''''
EXECIO 1 DISKW DIAGNOSE TABLE A 4 '(STRING' 5 6 '''',
  CENTER('FY',6) CENTER('RESIDUAL',16) CENTER('RELATIVE ACCURACY',36)''''
EXECIO 1 DISKW DIAGNOSE TABLE A 5 '(STRING' 6 6 '''',
  CENTER('--',6) CENTER('--------',16) CENTER('----------------',36)''''
LINE=6;PLACE=7
DO I = 4 TO N
   EXECIO 1 DISKW DIAGNOSE TABLE A LINE  '(STRING' PLACE 6 '''',
       CENTER(YEAR.I,6) CENTER(FORMAT(RES.I,8,0),16),
    CENTER(FORMAT(RAI.I,3,1)'%',36)''''
LINE=LINE+1;PLACE=PLACE+1
END
PLACE=PLACE+1
EXECIO 1 DISKW DIAGNOSE TABLE A LINE  '(STRING' PLACE 1 ' '
EXECIO 1 DISKW DIAGNOSE TABLE A LINE+1 '(STRING' PLACE+1  6 '''',
    CENTER(' ',10) CENTER('FY',6) CENTER('STABILITY INDEX',37)''''
EXECIO 1 DISKW DIAGNOSE TABLE A LINE+2 '(STRING' PLACE+2 6 '''',
    CENTER(' ',10) CENTER('--',6) CENTER('---------------',37)''''
LINE=LINE+3;PLACE=PLACE+3
DO I = N TO N+1
   YR = YEAR.1+I-1
EXECIO 1 DISKW DIAGNOSE TABLE A LINE  '(STRING' PLACE 6 '''',
  CENTER(' ',10) CENTER(YR,6) CENTER(FORMAT(SI.I,4,1)'%',36)''''
LINE=LINE+1;PLACE=PLACE+1
END
PLACE=PLACE+1
EXECIO 1 DISKW DIAGNOSE TABLE A LINE  '(STRING' PLACE 1 '   '
EXECIO 1 DISKW DIAGNOSE TABLE A LINE+1 '(STRING' PLACE+1 6 '''',
    CENTER(' ',10) CENTER('FY',6) CENTER('PRECISION INDEX',37)''''
EXECIO 1 DISKW DIAGNOSE TABLE A LINE+2  '(STRING' PLACE+2 6 '''',
    CENTER(' ',10) CENTER('--',6) CENTER('---------------',37)''''
LINE=LINE+3;PLACE=PLACE+3
DO I = 4 TO N+1
   YR = YEAR.1+I-1
EXECIO 1 DISKW DIAGNOSE TABLE A LINE  '(STRING' PLACE 6 '''',
    CENTER(' ',10)  CENTER(YR,6) CENTER(FORMAT(PI.I,5,1)'%',36)''''
LINE=LINE+1;PLACE=PLACE+1
END
EXECIO 1 DISKW DIAGNOSE TABLE A LINE  '(STRING' 24 7,
  '''PRESS PF12 FOR A LIST OF OPTIONS,PF2 TO EXIT.'''
DIAGNOSEBEGIN:
'DVHMEN DIAGNOSE TABLE A       '
PULL . . . KEY
CALL FUNCTION
RETURN
```

```
/*************************************************************************/
/* FUNCTION:SUBROUTINE TO DISPLAY COSTCASTER OUTPUTS */
/*************************************************************************/
FUNCTION:
IF KEY=PF10 THEN DO
        QUEUE '.ALARM'
        QUEUE 24 12 PH '''***PF10 NOT DEFINED       ***    '''
        CURSOR = ' 1 1 '
        SIGNAL MENU3
      END
IF KEY=PF09 THEN SIGNAL RESTART
IF KEY=PF01 THEN DO
        QUEUE '.ALARM'
        QUEUE 24 12 PH '''***OPTION NOT IMPLEMENTED***    '''
        CURSOR = ' 5 2 '
        SIGNAL MENU3
      END
IF KEY=PF02 THEN SIGNAL COSTCAT
IF KEY=PF07 THEN DO
        QUEUE '.ALARM'
        QUEUE 24 12 PH '''***OPTION NOT IMPLEMENTED***    '''
        CURSOR = '17 2 '
        SIGNAL MENU3
      END
IF KEY=PF11 THEN DO
        QUEUE '.ALARM'
        QUEUE 24 12 PH '''***PF KEY NOT IMPLEMENTED***    '''
        CURSOR = '1 1 '
        SIGNAL MENU3
      END
IF REGFLAG  = 1 THEN CALL REG
IF KEY=PF03 THEN SIGNAL PREDTABLE
IF KEY=PF04 THEN SIGNAL DIAGNOSE
IF KEY=PF05 THEN SIGNAL COSTAVOID
IF KEY=PF06 THEN SIGNAL WHENBUY
IF KEY=PF08 THEN SIGNAL LIFESAVE
IF KEY = PF12 THEN DO
CURSOR =' 1  1'
CALL MENU3
END
RETURN
```

```
/*************************************************************************/
/* INTLOG:  SUBROUTINE TO LOOK UP THE LOGS OF INTEGERS                   */
/*************************************************************************/
INTLOG:
ARG X
 IF X=  1 THEN RETURN 0.00000  ;IF X=  2 THEN RETURN 0.6931472
 IF X=  3 THEN RETURN 1.0986123;IF X=  4 THEN RETURN 1.3862944
 IF X=  5 THEN RETURN 1.6094379;IF X=  6 THEN RETURN 1.7917596
 IF X=  7 THEN RETURN 1.9459101;IF X=  8 THEN RETURN 2.0794416
 IF X=  9 THEN RETURN 2.1972246;IF X= 10 THEN RETURN 2.3026851
 IF X= 11 THEN RETURN 2.3978953;IF X= 12 THEN RETURN 2.4849066
 IF X= 13 THEN RETURN 2.5649494;IF X= 14 THEN RETURN 2.6390673
 IF X= 15 THEN RETURN 2.7080502;IF X= 16 THEN RETURN 2.7725887
 IF X= 17 THEN RETURN 2.8332133;IF X= 18 THEN RETURN 2.8903718
 IF X= 19 THEN RETURN 2.9444439;IF X= 20 THEN RETURN 2.9957323
 IF X= 21 THEN RETURN 3.0445224;IF X= 22 THEN RETURN 3.0910425
 IF X= 23 THEN RETURN 3.1354942;IF X= 24 THEN RETURN 3.1780538
 IF X= 25 THEN RETURN 3.2188758;IF X= 26 THEN RETURN 3.2580965
 IF X= 27 THEN RETURN 3.2958369;IF X= 28 THEN RETURN 3.3322045
 IF X= 29 THEN RETURN 3.3672958;IF X= 30 THEN RETURN 3.4011974
RETURN
```

```
/*********************************************************************/
/* OUTPUT THE DESIRED TABLE                                          */
/*********************************************************************/
TABLEPUT:
PROCEDURE EXPOSE PREDFLAG WHENFLAG COSTFLAG LIFEFLAG DIAGFLAG COSTCATFLAG
IF PREDFLAG=1 THEN DO I=1 TO 48
EXECIO 1 DISKR PREDTABLE TABLE A I
IF RC=2 THEN LEAVE
PULL LEAD "'" OUTPUT "'" TRAIL
TRACE OFF
IF POS('PRESS PF',OUTPUT) =0 THEN  EXECIO 1 DISKW PREDTABLE LISTING A I
'(STRING' OUTPUT
END
IF PREDFLAG=1 THEN 'LISTOFF PREDTABLE LISTING A (VERSATEC)'
J=1
IF WHENFLAG=1 THEN DO I=1 TO 48
EXECIO 1 DISKR WHENBUY TABLE A I
IF RC=2 THEN LEAVE
PULL LEAD "'" OUTPUT "'" TRAIL
TRACE OFF
IF POS('PRESS PF',OUTPUT) =0 THEN DO
 EXECIO 1 DISKW WHENBUY LISTING A J '(STRING' OUTPUT
 J=J+1
END
END
IF WHENFLAG=1 THEN 'LISTOFF WHENBUY LISTING A   (VERSATEC)'
IF COSTFLAG=1 THEN DO I=1 TO 48
EXECIO 1 DISKR COSTAVOID TABLE A I
IF RC=2 THEN LEAVE
PULL LEAD "'" OUTPUT "'" TRAIL
IF POS('PRESS PF',OUTPUT) =0 THEN EXECIO 1 DISKW COSTAVOID LISTING A I
'(STRING' OUTPUT
END
IF COSTFLAG=1 THEN 'LISTOFF COSTAVOID LISTING A (VERSATEC)'
IF LIFEFLAG=1 THEN DO I=1 TO 48
EXECIO 1 DISKR LIFESAVE TABLE A I
IF RC=2 THEN LEAVE
PULL LEAD "'" OUTPUT "'" TRAIL
IF POS('PRESS PF',OUTPUT) =0 THEN EXECIO 1 DISKW LIFESAVE LISTING A I
'(STRING' OUTPUT
END
IF LIFEFLAG=1 THEN 'LISTOFF LIFESAVE LISTING A  (VERSATEC)'
IF DIAGFLAG=1 THEN DO I=1 TO 48
EXECIO 1 DISKR DIAGNOSE TABLE A I
IF RC=2 THEN LEAVE
PULL LEAD "'" OUTPUT "'" TRAIL
IF POS('PRESS PF',OUTPUT) =0 THEN EXECIO 1 DISKW DIAGNOSE LISTING A I
'(STRING' OUTPUT
END
IF DIAGFLAG=1 THEN 'LISTOFF DIAGNOSE LISTING A  (VERSATEC)'
IF COSTCATFLAG=1 THEN DO I=1 TO 24
EXECIO 1 DISKR COSTCAT TABLE A I
IF RC=2 THEN LEAVE
PULL LEAD "'" OUTPUT "'" TRAIL
IF POS('PRESS PF',OUTPUT) =0 THEN EXECIO 1 DISKW COSTCAT LISTING A I
'(STRING' OUTPUT
END
IF COSTCATFLAG=1 THEN 'LISTOFF COSTCAT  LISTING A  (VERSATEC)'
RETURN
```

```
/*********************************************************************/
/* COSTCAT: SUBROUTINE TO PRODUCE THE TABLE OF INPUT COSTS  */
/*          BY COST CATEGORY.                                        */
/*********************************************************************/
COSTCAT:
COSTCATFLAG=1
CALL COSTREAD
I=1;J=2;K=3;L=4;M=5
EXECIO 1 DISKW COSTCAT TABLE A 1 '(STRING' 1 1 '''',
CENTER(TMS NOMEN.N 'INPUT TABLE(COSTS ARE $/UNIT IN FY82 $)',64) ''''
EXECIO 1 DISKW COSTCAT TABLE A 2 '(STRING' 2 1 '''',
CENTER('COST CATEGORY',24) CENTER('FY'YEAR.I,7) CENTER('FY'YEAR.I+1,7),
CENTER('FY'YEAR.I+2,7) CENTER('FY'YEAR.I+3,7) CENTER('FY'YEAR.I+4,7)''''
EXECIO 1 DISKW COSTCAT TABLE A 3 '(STRING' 3 1 '''',
CENTER('------------',24) CENTER('--------',7) CENTER('----------',7),
CENTER('----------',7) CENTER('----------',7) CENTER('----------',7)''''
EXECIO 1 DISKW COSTCAT TABLE A 4 '(STRING' 4 1 '''',
CENTER('OPERATIONS PERSONNEL     ',24) CENTER(FORMAT(COST.1.I,5,0),7)
CENTER(FORMAT(COST.1.J,5,0),7),
CENTER(FORMAT(COST.1.K,5,0),7) CENTER(FORMAT(COST.1.L,5,0),7)
CENTER(FORMAT(COST.1.M,5,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 5 '(STRING' 5 1 '''',
CENTER('BASE MAINTENANCE PERS.   ',24) CENTER(FORMAT(COST.2.I,5,0),7)
CENTER(FORMAT(COST.2.J,5,0),7),
CENTER(FORMAT(COST.2.K,5,0),7) CENTER(FORMAT(COST.2.L,5,0),7)
CENTER(FORMAT(COST.2.M,5,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 6 '(STRING' 6 1 '''',
CENTER('ADMINISTRATIVE PERSONNEL ',24) CENTER(FORMAT(COST.3.I,5,0),7)
CENTER(FORMAT(COST.3.J,5,0),7),
CENTER(FORMAT(COST.3.K,5,0),7) CENTER(FORMAT(COST.3.L,5,0),7)
CENTER(FORMAT(COST.3.M,5,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 7 '(STRING' 7 1 '''',
CENTER('SUPPLY SUPPORT PERSONNEL ',24) CENTER(FORMAT(COST.4.I,5,0),7)
CENTER(FORMAT(COST.4.J,5,0),7),
CENTER(FORMAT(COST.4.K,5,0),7) CENTER(FORMAT(COST.4.L,5,0),7)
CENTER(FORMAT(COST.4.M,5,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 8 '(STRING' 8 1 '''',
CENTER('FUEL                     ',24) CENTER(FORMAT(COST.5.I,5,0),7)
CENTER(FORMAT(COST.5.J,5,0),7),
CENTER(FORMAT(COST.5.K,5,0),7) CENTER(FORMAT(COST.5.L,5,0),7)
CENTER(FORMAT(COST.5.M,5,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 9 '(STRING' 9 1 '''',
CENTER('MAINTENANCE MATERIEL     ',24) CENTER(FORMAT(COST.6.I,5,0),7)
CENTER(FORMAT(COST.6.J,5,0),7),
CENTER(FORMAT(COST.6.K,5,0),7) CENTER(FORMAT(COST.6.L,5,0),7)
CENTER(FORMAT(COST.6.M,5,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 10 '(STRING' 10 1 '''',
CENTER('UTILITIES                ',24) CENTER(FORMAT(COST.7.I,5,0),7)
CENTER(FORMAT(COST.7.J,5,0),7),
CENTER(FORMAT(COST.7.K,5,0),7) CENTER(FORMAT(COST.7.L,5,0),7)
CENTER(FORMAT(COST.7.M,5,0),7)''''
```

```
EXECIO 1 DISKW COSTCAT TABLE A 11 '(STRING' 11 1 '''',
CENTER('DEPOT MAINTENANCE          ',24) CENTER(FORMAT(COST.8.I,6,0),7)
CENTER(FORMAT(COST.8.J,6,0),7),
CENTER(FORMAT(COST.8.K,6,0),7) CENTER(FORMAT(COST.8.L,6,0),7)
CENTER(FORMAT(COST.8.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 12 '(STRING' 12 1 '''',
CENTER('REPLACEMENT INVESTMENT     ',24) CENTER(FORMAT(COST.9.I,6,0),7)
CENTER(FORMAT(COST.9.J,6,0),7),
CENTER(FORMAT(COST.9.K,6,0),7) CENTER(FORMAT(COST.9.L,6,0),7)
CENTER(FORMAT(COST.9.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 13 '(STRING' 13 1 '''',
CENTER('BASE OPERATIONS SUPPORT    ',24) CENTER(FORMAT(COST.10.I,6,0),7)
CENTER(FORMAT(COST.10.J,6,0),7),
CENTER(FORMAT(COST.10.K,6,0),7) CENTER(FORMAT(COST.10.L,6,0),7)
CENTER(FORMAT(COST.10.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 14 '(STRING' 14 1 '''',
CENTER('REAL PROPERTY MAINT.       ',24) CENTER(FORMAT(COST.11.I,6,0),7)
CENTER(FORMAT(COST.11.J,6,0),7),
CENTER(FORMAT(COST.11.K,6,0),7) CENTER(FORMAT(COST.11.L,6,0),7)
CENTER(FORMAT(COST.11.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 15 '(STRING' 15 1 '''',
CENTER('COMMUNICATIONS             ',24) CENTER(FORMAT(COST.12.I,6,0),7)
CENTER(FORMAT(COST.12.J,6,0),7),
CENTER(FORMAT(COST.12.K,6,0),7) CENTER(FORMAT(COST.12.L,6,0),7)
CENTER(FORMAT(COST.12.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 16 '(STRING' 16 1 '''',
CENTER('TEMPORARY DUTY             ',24) CENTER(FORMAT(COST.13.I,6,0),7)
CENTER(FORMAT(COST.13.J,6,0),7),
CENTER(FORMAT(COST.13.K,6,0),7) CENTER(FORMAT(COST.13.L,6,0),7)
CENTER(FORMAT(COST.13.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 17 '(STRING' 17 1 '''',
CENTER('PERM. CHANGE OF STATION    ',24) CENTER(FORMAT(COST.14.I,6,0),7)
CENTER(FORMAT(COST.14.J,6,0),7),
CENTER(FORMAT(COST.14.K,6,0),7) CENTER(FORMAT(COST.14.L,6,0),7)
CENTER(FORMAT(COST.14.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 18 '(STRING' 18 1 '''',
CENTER('MEDICAL                    ',24) CENTER(FORMAT(COST.15.I,6,0),7)
CENTER(FORMAT(COST.15.J,6,0),7),
CENTER(FORMAT(COST.15.K,6,0),7) CENTER(FORMAT(COST.15.L,6,0),7)
CENTER(FORMAT(COST.15.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 19 '(STRING' 19 1 '''',
CENTER('GENERAL DEPOT SUPPORT      ',24) CENTER(FORMAT(COST.16.I,6,0),7)
CENTER(FORMAT(COST.16.J,6,0),7),
CENTER(FORMAT(COST.16.K,6,0),7) CENTER(FORMAT(COST.16.L,6,0),7)
CENTER(FORMAT(COST.16.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 20 '(STRING' 20 1 '''',
CENTER('ENGINEERING SUPPORT        ',24) CENTER(FORMAT(COST.17.I,6,0),7)
CENTER(FORMAT(COST.17.J,6,0),7),
CENTER(FORMAT(COST.17.K,6,0),7) CENTER(FORMAT(COST.17.L,6,0),7)
CENTER(FORMAT(COST.17.M,6,0),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 21 '(STRING' 21 1 '''',
CENTER('TRANSPORT. AND PACKAGING',24) CENTER(FORMAT(COST.18.I,6,0),7)
CENTER(FORMAT(COST.18.J,6,0),7),
CENTER(FORMAT(COST.18.K,6,0),7) CENTER(FORMAT(COST.18.L,6,0),7)
CENTER(FORMAT(COST.18.M,6,0),7)''''
```

```
EXECIO 1 DISKW COSTCAT TABLE A 22 '(STRING' 22 1 '''',
CENTER('ADVANCED TRAINING          ',24) CENTER(FORMAT(COST.19.I,6,Ø),7)
CENTER(FORMAT(COST.19.J,6,Ø),7),
CENTER(FORMAT(COST.19.K,6,Ø),7) CENTER(FORMAT(COST.19.L,6,Ø),7)
CENTER(FORMAT(COST.19.M,6,Ø),7)''''
EXECIO 1 DISKW COSTCAT TABLE A 23 '(STRING' 23 1 '''',
                                                    ''''
EXECIO 1 DISKW COSTCAT TABLE A 24 '(STRING' 24 1 '''
               PRESS PF2 TO EXIT,PF12 FOR OTHER OPTIONS . '''
'DVHMEN COSTCAT TABLE A'
PULL . . . KEY
IF KEY = PFØ2 THEN RETURN
CALL FUNCTION
RETURN
```

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: 118-8	2. GOVT ACCESSION NO.: AD-A161 636
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): PROTOTYPE IMPLEMENTATION OF COSTCASTER, A COST-PREDICTION AND TRADE-OFF MODEL FOR AIR FORCE GROUND C-E EQUIPMENT	5. TYPE OF REPORT & PERIOD COVERED: Technical Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Gregory J. Zunic, Dennis E. Smith, Kevin C. Burns, Robert L. Gardner	8. CONTRACT OR GRANT NUMBER(s): F33600-82-C-0466
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Desmatics, Inc., P.O. Box 618, State College, PA 16804	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: HQ AFLC/MML(VAMOSC), Wright-Patterson AFB, OH 45433	12. REPORT DATE: July 1985
	13. NUMBER OF PAGES: 69
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this report is unlimited.

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Cost Prediction
Trade-Off Analysis
O&S Costs
VAMOSC
Ground C-E Equipment

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

COSTCASTER is a computer-based system developed by Desmatics, Inc. for the U.S. Air Force. Its purpose is to aid in making cost-related decisions regarding replacement, modification or retention of Air Force ground Communications-Electronics (C-E) equipments. This report includes a brief summary of the COSTCASTER methodology, which is described more fully in a previous Desmatics report. However, the major emphasis here is to document the implementation of a prototype computer model developed to demonstrate the feasibility of the COSTCASTER concept.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

END

FILMED

1-86

DTIC